[10] Simulation of Peptide Prohormone Processing and Peptidergic Granule Transport and Release in Neurosecretory Cells

Daniel K. Hartline, Robert Whitney Newcomb, and Robert Wayne Newcomb

Introduction

Peptidergic secretion offers a potential degree of subtlety and complexity not possible with classic small molecule neurotransmitters and hormone systems. Bioactive peptides are synthesized as part of larger amino acid chains, which may contain a number of bioactive amino acid sequences. The prohormone is packaged into secretory granules, which serve for the storage and release of the peptides. Within the secretory granules, enzymatic cleavage of the prohormone into active peptides occurs [for reviews, see Douglass *et al.* (1); Hokfelt (2)]. Consequently, such cells release multiple chemical messengers in response to a single stimulus, potentially permitting a single neural or endocrine cell to control a coordinated response to its secretion (Fig. 1).

It is experimentally observed that the degree of proteolytic cleavage (and thus the spectrum of peptides potentially released) often varies with cellular location and physiological state (3–6). Both the complexity of the chemical mixture as well as its changing character with physiological state are potentially important features of peptidergic secretion. These properties contrast with classic transmitter systems [for review, see Kanner and Schuldiner (7)], whose behavior is, despite kinetic complexity, limited to an increase or decrease in the rate of secretion of one or several bioactive compounds. Schematic representations contrasting these two types of bioactive compounds are summarized in Fig. 2. Given the complexity of the peptidergic cell, methods which can describe its dynamic properties should help in elucidating the mechanisms by which generation and release of mixtures of bioactive peptides are controlled. This in turn will help further the understanding of how individual peptide-secreting cells function in neural and endocrine networks.

To facilitate the quantitative analysis of these complexities, this chapter

Methods in Neurosciences, Volume 28

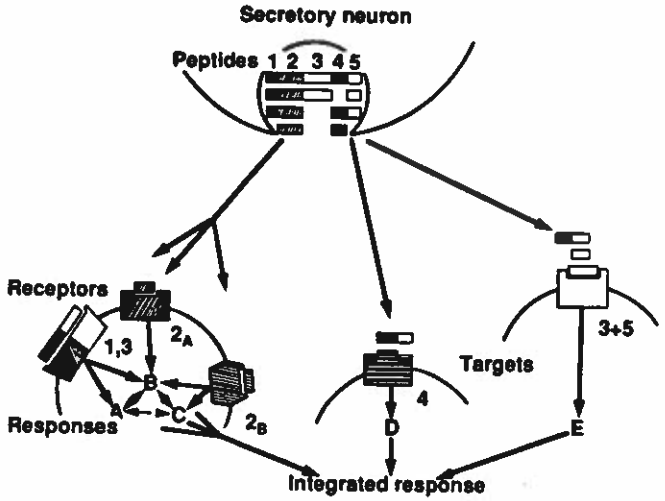

FIG. 1 Potential complexity of the response to peptidergic secretion. The partial proteolytic cleavage of a propeptide with multiple similar and distinct sequences is illustrated. Interaction of the released mixture with multiple receptors on distinct cell types is diagrammed.

develops several computer models and mathematical methods which provide the following capabilities. First, they provide analysis of the kinetics of the proteolytic cleavage of a prohormone. This will include the determination of the rate constants of proteolytic processing from biochemical measurements. Although outside of the scope of this chapter, there are a number of newer experimental approaches to the study of prohormone processing which, in combination with classic pulse–chase experiments, should be amenable to the type of quantitative analysis developed here (8–12). Second, the models provide simulation of the behavior of a peptidergic secretory system (i.e., computer models of peptide synthesis, processing, transport, storage, and release).

Kinetics of Proteolytic Cleavage of Prohormone

When a prohormone is processed into peptide fragments, there is in general a set of specific potential cleavage sites, each acted on by one or more specific enzymes. As the prohormone and its fragments are progressively attacked, more cleavage sites may become exposed, and/or the affinity of previously exposed sites for enzymes (and hence the rate constant for cleav-

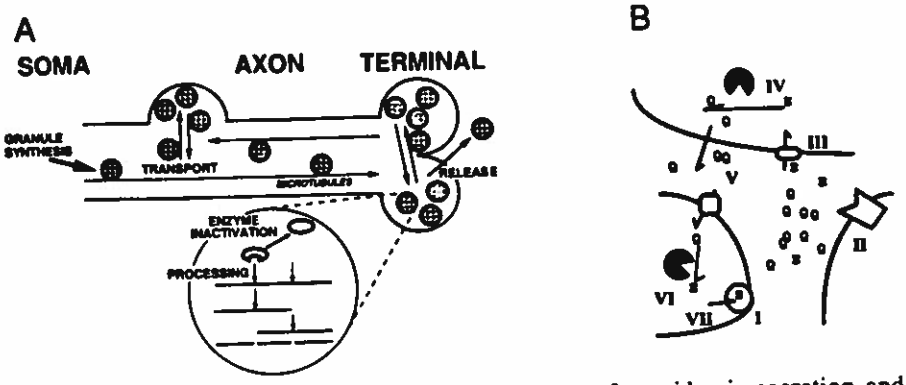

FIG. 2 Schematic comparison of cellular properties of peptidergic secretion and classic secretion of small molecule neurotransmitters. (A) Peptidergic system, with transport, storage, and release of secretory granules in different axonal compartments, proteolytic cleavage of propeptides in secretory granules, and potential changes in enzymatic activity within secretory granules. Further detail is provided in the text [modified from Hartline and Newcomb (13)]. (B) Classic system: small molecule transmitters. Glutamate (E) is used as a specific example. The neurotransmitter is released into the synaptic cleft from storage vesicles (I), where it acts on its receptors (II) and is then removed from the cleft by a specific transporter (III). The active molecule is enzymatically converted (IV) to the inactive compound glutamine (Q) in nonneuronal cells. Glutamine is released into the extracellular fluid and transported into the neuron (V), where it is enzymatically converted back to glutamate (VI) and transported back into the vesicle (VII).

age) may change. In this section we discuss how this situation may be represented mathematically so that the time course of rise and fall of various peptide products and intermediates may be calculated.

1. A prohormone is represented as a chain of peptide segments representing the fragments produced by complete cleavage at all potential sites on the prohormone. Cleavage sites between segments are numbered 1 through N, starting at the amino-terminal end (left end in the figure schematics). Segments are identified by the number of the cleavage site to their right (the final segment is designated $N + 1$). This situation is diagrammed in Fig. 3.

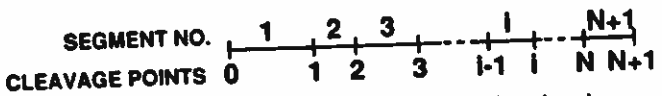

FIG. 3 Schematic representation of a prohormone showing the cleavage sites, peptide segments, and numbering system used for the cleavage model.

2. $P_{i,j}$ will represent a peptide fragment (or the amount or concentration of such) including that of segments i through j. $P_{i,i}$ (also abbreviated P_i) is the single ith segment.

3. $K_{i,j,m}$ will be the first-order nonnegative rate constant of cleavage of fragment $P_{i,j}$ at site m (higher order reactions are not considered in this treatment).

4. A crucial assumption is that the system initially starts, at $t = 0$, with $P_{1,N+1}$ of nonzero concentration P_0 and all others zero.

Basic Equations

For each fragment, $P_{i,j}$, the amount present is determined by a combination of reactions which (1) generate that fragment from precursors (two categories of reaction do this: cleavages at site $i - 1$ of all fragments ending at site j and cleavages at site j of all fragments beginning at site $i - 1$) and (2) degrade it by cleavage at any internal site m (with $i \leq m < j$). For first-order reactions this means (6)

$$dP_{i,j}/dt = \sum_{m=1}^{i-1} K_{m,j,i-1} P_{m,j} + \sum_{m=j+1}^{N+1} K_{i,m,j} P_{i,m} - \left(\sum_{m=1}^{j-1} K_{i,j,m} \right) P_{i,j},$$

$$\text{for } i < j \text{ over } 1, 2, \ldots, N+1. \tag{1}$$

Stoichiometry constrains this set of equations so that, whereas there are $(N + 1)(N + 2)/2$ peptides (including the precursor) in the mixture (13), only $N(N + 1)/2$ are independently variable. The sum of all peptides containing a particular segment must equal the amount of prohormone originally present, whence

$$P_{i,i} = P_0 - \sum_{m=1}^{i-1} P_{m,i} - \sum_{j=i+1}^{N+1} P_{i,j}. \tag{2}$$

Note that Eq. (2) reflects the fact that the final concentration of 1-segment products, $P_{i,i}$, approaches the initial concentration of the prohormone, since all of the multisegment intermediates are degraded over time and hence $P_{i,j} \rightarrow 0$ for i not equal to j.

Solutions to Basic Equations for N = 3

We now derive the mathematical form describing the time course of buildup and decay of the remaining peptides of Eq. (1). To express the concepts efficiently, matrix notation will be used, for which an excellent reference is

the text of Searle (14). The equations describing the cleavage processes are expressed as ordinary differential equations for which an outstanding and geometrically oriented reference is the text of Arnold (15).

To best illustrate the state of affairs let us choose a three-cleavage-site case ($N = 3$). First we write out Eq. (1) and collect the results into the following matrix form using a convenient ordering of the $P_{i,j}$ into a vector $\mathbf{P}$:

$$\frac{d}{dt}\begin{bmatrix} P_{1.4} \\ P_{1.3} \\ P_{1.2} \\ P_{2.4} \\ P_{2.3} \\ P_{3.4} \end{bmatrix} = \begin{bmatrix} -K_{1.4.1} - K_{1.4.2} - K_{1.4.3} & 0 & 0 & 0 & 0 & 0 \\ K_{1.4.3} & -K_{1.3.1} - K_{1.3.2} & 0 & 0 & 0 & 0 \\ K_{1.4.2} & K_{1.3.2} & -K_{1.2.1} & 0 & 0 & 0 \\ K_{1.4.1} & 0 & 0 & -K_{2.4.2} - K_{2.4.3} & 0 & 0 \\ 0 & K_{1.3.1} & 0 & K_{2.4.3} & -K_{2.3.2} & 0 \\ K_{1.4.2} & 0 & 0 & K_{2.4.2} & 0 & -K_{3.4.3} \end{bmatrix} \begin{bmatrix} P_{1.4} \\ P_{1.3} \\ P_{1.2} \\ P_{2.4} \\ P_{2.3} \\ P_{3.4} \end{bmatrix}. \quad (3a)$$

By simply renaming the elements of the large matrix as A_{ij}s we see that this last equation has the nice lower triangular structure

$$\frac{d}{dt}\begin{bmatrix} P_{1.4} \\ P_{1.3} \\ P_{1.2} \\ P_{2.4} \\ P_{2.3} \\ P_{3.4} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 & 0 & 0 & 0 \\ A_{21} & A_{22} & 0 & 0 & 0 & 0 \\ A_{31} & A_{32} & A_{33} & 0 & 0 & 0 \\ A_{41} & 0 & 0 & A_{44} & 0 & 0 \\ 0 & A_{52} & 0 & A_{54} & A_{55} & 0 \\ A_{61} & 0 & 0 & A_{64} & 0 & A_{66} \end{bmatrix} \begin{bmatrix} P_{1.4} \\ P_{1.3} \\ P_{1.2} \\ P_{2.4} \\ P_{2.3} \\ P_{3.4} \end{bmatrix}, \quad (3b)$$

whereas the relabeling is recorded in the 6×6 matrix quality

$$\mathbf{A} = \begin{bmatrix} -K_{1.4.1} - K_{1.4.2} - K_{1.4.3} & 0 & 0 & 0 & 0 & 0 \\ K_{1.4.3} & -K_{1.3.1} - K_{1.3.2} & 0 & 0 & 0 & 0 \\ K_{1.4.2} & K_{1.3.2} & -K_{1.2.1} & 0 & 0 & 0 \\ K_{1.4.1} & 0 & 0 & -K_{2.4.2} - K_{2.4.3} & 0 & 0 \\ 0 & K_{1.3.1} & 0 & K_{2.4.3} & -K_{2.3.2} & 0 \\ K_{1.4.2} & 0 & 0 & K_{2.4.2} & 0 & -K_{3.4.3} \end{bmatrix}. \quad (4)$$

that is, $A_{11} = -(K_{1,4,1} + K_{1,4,2} + K_{1,4,3})$, etc., as is seen by equating the coefficients in Eq. (3b) with those in Eq. (3a).

The solution to Eqs. (3a) and (3b) may be obtained using convolution, the fundamental operation of linear time-invariant systems theory (16), for which an extensive mathematical treatment exists (17). Because of the nice form of the state equations, an explicit form for the solution can be given using the convolution integral, this latter, denoted as $a * b$, being defined for two time functions $a(\cdot)$ and $b(\cdot)$ by the rather messy looking but extremely useful integral

$$a(\cdot) * b(\cdot)(t) = \int_0^t a(t-\tau)b(\tau)\,d\tau \qquad \text{for } t > 0. \tag{5}$$

In our case all convolutions essentially reduce to those of two exponentials so the result is quite convenient, being

$$e^{\alpha t} * e^{\beta t} = \begin{cases} \dfrac{e^{\beta t} - e^{\alpha t}}{\beta - \alpha} & \text{if } \beta \neq \alpha \\ te^{\alpha t} & \text{if } \beta = \alpha. \end{cases} \tag{6}$$

Further, in proceeding down the rows of Eq. (3a) or (3b) after the first row, we will only need to solve equations of the form

$$\frac{dx}{dt} = f(t) + \alpha x \qquad x(0) = 0, \tag{7}$$

for which the convolution solution is

$$x(t) = e^{\alpha x} * f(t). \tag{8}$$

In the case of $N = 3$, $P_{1,4}$ is first found by direct integration of the first row of Eq. (3b), then used in the convolutions to obtain the other fragments, as follows:

$$P_{1,4} = e^{A_{11}t}P_0,$$

$$P_{1,3} = e^{A_{22}t} * A_{21}P_{1,4} = e^{A_{22}t} * A_{21}e^{A_{11}t}P_0 = \begin{cases} \dfrac{e^{A_{22}t} - e^{A_{11}t}}{A_{22} - A_{11}}A_{21}P_0 & \text{if } A_{22} \neq A_{11} \\ te^{A_{11}t}A_{21}P_0 & \text{if } A_{22} = A_{11} \end{cases},$$

$$P_{1,2} = e^{A_{33}t} * (A_{31}P_{1,4} + A_{32}P_{1,3}) = e^{A_{33}t} * (A_{31} + A_{32}e^{A_{22}t}A_{21}) * e^{A_{11}t}P_0, \tag{9}$$

$$P_{2,4} = e^{A_{44}t} * A_{41}P_{1,4} = e^{A_{44}t} * A_{41}e^{A_{11}t}P_0,$$

$$P_{2,3} = e^{A_{55}t} * (A_{52}P_{1,3} + A_{54}P_{2,4}) = e^{A_{55}t} * (A_{52}e^{A_{22}t}A_{21} + A_{54}e^{A_{44}t}A_{41}) * e^{A_{11}t}P_0, \quad (10)$$

$$P_{3,4} = e^{A_{66}t} * (A_{61}P_{1,4} + A_{64}P_{2,4}) = e^{A_{55}t} * (A_{61} + A_{64}e^{A_{44}t}A_{41}) * e^{A_{11}t}P_0. \quad (11)$$

As seen by these equations, the convolution approach views the time course of a particular intermediate as the result of a simple first-order drain in the peptide concentration at an aggregate rate constant equal to the sum of all of the first-order rate constants for cleavage of that particular intermediate [the last term in Eq. (1)]. This simple exponential is driven by (convolved with) the, in general, complex, time course of production of that intermediate from all of its various immediate precursors [Eq. (8)]. Because of the nice form of the original cleavage equation, Eq. (1), Eqs. (9)–(11) are readily extended for any N.

An example of the kinetics just described, which shows as well some of the difficulties in the interpretation of the results of experiments which look at prohormone processing, is illustrated in Fig. 4. This figure considers two different kinetic situations which might arise in the processing of a

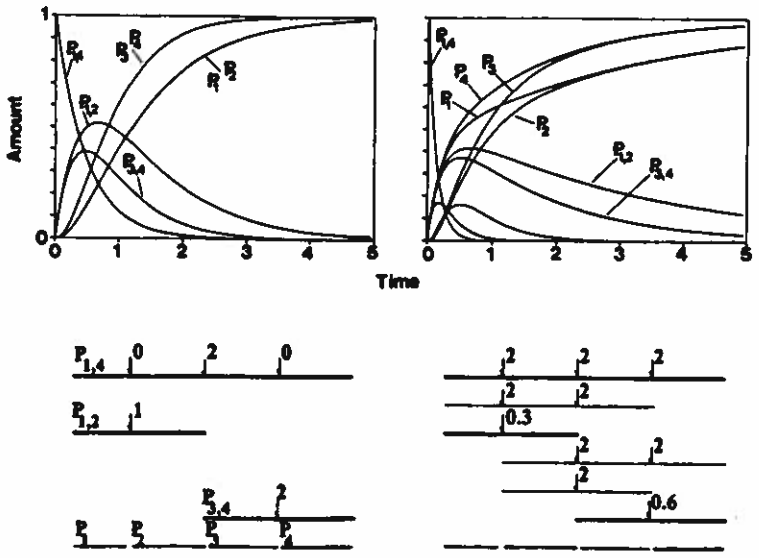

FIG. 4 Simulated processing of model prohormones by sequential (left) and simultaneous (right) mechanisms. Numbers indicate cleavage rate constants in arbitrary reciprocal time units. Intermediates which accumulate significantly are emphasized with thick lines in the diagrams at bottom.

prohormone containing three internal cleavage sites. In the graph at the left, an ordered processing reaction involving peptide-length-dependent cleavage sites is shown. In this example, all rate constants are set to 0, except for $K_{1,4,2} = 2.0$ on $P_{1,4}$, $K_{3,4,3} = 2.0$ on $P_{3,4}$, and $K_{1,2,1} = 1.0$ on $P_{1,2}$, where rate constants are in arbitrary time units (e.g., days^{-1}). The decay of the prohormone from this mechanism is contrasted on the right-hand side of Fig. 4 to a situation in which multiple processing pathways occur. In the example shown, rapid cleavage (K terms of 2.0) occurs at all but several cleavage sites on selected intermediates ($K_{1,2,1} = 0.3$ on $P_{1,2}$, and $K_{3,4,3} = 0.6$ on $P_{3,4}$). In this latter case, much of the biosynthesis of the final product peptides occurs through a variety of intermediates that do not accumulate to a significant extent. However, both this and the length-dependent cleavage rate situation involve the accumulation of a limited number of intermediates in significant quantity. Qualitative description of prohormone processing, particularly if coupled with low resolution methods of chemical analysis, will not be able to distinguish between these two different situations.

General Solution to the Basic Equations

Equation (1) represents linear time-invariant ordinary differential equations with no input except the one initial condition P_0. Consequently they can be rewritten in state-vector form as

$$\frac{d\mathbf{P}}{dt} = \mathbf{A} \cdot \mathbf{P}, \tag{12}$$

where $\mathbf{P}$ is a column vector of the $P_{i,j}$, terms, called the state variable in systems theory nomenclature, and $\mathbf{A}$ a square matrix with its entries being linear combinations of the $K_{i,j,m}$, as the above example illustrates. The solution of Eq. (12) is readily obtained and is (see p. 106 in Ref. 15):

$$\mathbf{P(t)} = [\exp(\mathbf{A} \cdot t)] \cdot \mathbf{P(0)}, \tag{13}$$

where various methods are available for evaluating the matrix exponential (see pp. 123, 137, and 167 in Ref. 15); see also Eqs. (9)–(11) above for the form of most use to the equations on hand. Although this solution [Eq. (13)] is simply expressed, the simplicity is somewhat deceptive as it shows that estimation from this form of the solution involves nonlinear (exponential) functions of the K terms. For this reason we attempt to take advantage of the structure inherent in the problem.

By properly ordering the entries in the state vector **P** a particularly nice form for **A** results, this being lower triangular and conveniently partitioned if the first $N + 1$ entries of **P** go with $i = 1$ and inversely ordered numerically with j, the second N entries going with $i = 2$, and again inversely ordered numerically with j, the third $N - 1$ entries going with $i = 3$, etc. The column vector **P** is thus chosen as [written as a row vector through use of the matrix transpose (see p. 38 in Ref. 14), which changes a row vector into a column vector]

$$\begin{aligned}\mathbf{P} &= [P_{1,N+1}, P_{1,N}, \ldots, P_{1,2}; P_{2,N+1}, \ldots, P_{2,3}; \ldots; P_{N-1,N+1}, P_{N-1,N}; P_{N,N+1}]^{\text{transpose}},\\ \mathbf{P}(0) &= [P_0, 0, \ldots, 0]^{\text{transpose}},\end{aligned} \tag{14}$$

which agrees with the number given above (15). From this we see that the number of entries in **P** is $N(N + 1)/2 = N + (N - 1) + \ldots + 1$ terms total, which is also the number of rows and columns in **A**.

We note from our example above that the **A** matrix is indeed lower triangular, the advantage of which is that each row is dependent only on the rows above it and not those below. This allows us to start a solution from the top row and work down to the bottom row, finding the unknown entries of **A** of a given row at each step using the convolution on all but the first row, as shown in the example above.

Finding Rate Constants

There are a number of means available for finding the rate constants $K_{i,j,k}$ from **P(t)** though not all of them are equally effective, primarily due to numerical inaccuracies and other sources of calculation and measurement noise. For those interested, there are commercial programs which can be generally used though usually requiring considerable learning to be effective. For those interested in writing their own program, by which probably more insight can be gained, the curve fitting method based on the exact solution via convolution is recommended [see Eqs. (21)–(25)]. The method is actually presented last since it relies on a number of developments including least-squares and related fittings. Indeed almost all practical techniques seem to require some sort of least-squares fitting, so that aspect is emphasized. For convenience the techniques presented here are summarized in Table I.

Since most parameter determination methods will first find the A terms we need to determine the K terms given the A's. We again use the example for $N = 3$, for which the 14 nonzero equations of Eq. (4) (only 10 of which

TABLE I Summary of Techniques

	Technique; subequations	Comments
Main equations	$\mathbf{K} = \mathbf{L_K A}$	Linear transformation of **A**; see (15)
1. $\mathbf{A} = \Theta[d\Theta/dt]^{-1}$	Θ represents columns of **P**(t)	Numerically inaccurate
2. $\mathbf{A} = I\Theta[\Theta]^{-1}$	$I\Theta$ is the numerical integral of Θ	Slightly improved
3. Symbolic solution [$\mathbf{At} = \ln \mathbf{P}(t) - \ln \mathbf{P}(0)$]		Exact; needs precise data
Least-squares fits		
4. $\mathbf{P}_e(t) = e^{\mathbf{A}_e t} \cdot \mathbf{P}(0)$		Good; messy equations
5. $\mathbf{P}_e(t) = \mathbf{A}_e I\mathbf{P}(t) + \mathbf{P}(0)$	$I\mathbf{P}$ is the numerical integral of **P**; see Eq. (16)	Satisfactory
6. $\mathbf{P}_e(t) = \mathbf{B}_e E\mathbf{P}(t)$ $\mathbf{A}_e = \mathbf{L}_A \mathbf{B}_e$	Curve fitting using convolution exponentials; $E\mathbf{P}$ represents columns of exponentials; see Eq. (24) for $\mathbf{L}_A$	Accurate for experimental data
Commercial programs		
SCIENTIST	Least-squares fit to ordinary differential equation $d\mathbf{P}/dt = \mathbf{A}(\mathbf{K}) \cdot \mathbf{P}$	Accurate, reasonably fast, easy to use
MatLab	Matrix solutions of all of above	Good; need to learn its language
Matlab system identification toolbox	General parameter estimation	Generic estimation package for MatLab language

are independent) are solved for the K's in terms of the A's, resulting in the following transformation (which can also be written in matrix form as the transformation is linear, but which is avoided here because it requires introducing other complications such as an appropriate numbering of the K's):

$$\begin{aligned}
&K_{1,4,1} = -A_{11} - A_{21} - A_{31} = A_{41}, && K_{2,4,2} = -A_{44} - A_{55} = A_{64},\\
&K_{1,4,2} = A_{31}, && K_{2,4,3} = A_{45},\\
&K_{1,4,3} = A_{21}, && K_{2,3,2} = -A_{55},\\
&K_{1,3,1} = -A_{22} - A_{32} = A_{52}, && K_{3,4,3} = -A_{66}.\\
&K_{1,3,2} = A_{32}, &&\\
&K_{1,2,1} = -A_{33}, &&
\end{aligned} \tag{15}$$

It should be noticed that there is dependence among the A_{ij} terms, though the $K_{i,j,m}$ are themselves generally independent.

The most obvious technique is to use $\mathbf{P}(t)$ at $N \times (N + 1)/2$ different times using them as columns in a square matrix Θ; placing their derivatives in $d\Theta/dt$ allows $\mathbf{A}$ to be solved via $\mathbf{A} = \Theta[d\Theta/dt]^{-1}$ where the superscript -1 denotes the matrix inverse (see p. 83 in Ref. 14). Unfortunately, the numerical formation of $d\Theta/dt$ is a very noisy process, something which can be circumvented by numerically integrating $\mathbf{P}(t)$; using the integral $I\mathbf{P}(t)$ leads to the integral $I\Theta$ of Θ, in which case, again using Eq. (13), $\mathbf{A} = I\Theta[\Theta]^{-1}$. Because of the lower triangular structure of $\mathbf{A}$ one can proceed in blocks, one block for each i of the P_{ij}, something which allows some alleviation of the numerical problems of forming the inverse. But the results have not proven to be very satisfying in practice.

A more satisfying solution is to use least-squares estimation techniques (see p. 228 in Ref. 14) on Eq. (3a) rewritten in integral form, $\mathbf{P} = \mathbf{A} \cdot I\mathbf{P} + \mathbf{P}(0)$. In this technique we take an estimated $\mathbf{A}$ (call it $\mathbf{A}_e$), and using $I\mathbf{P}$, obtain an estimated $\mathbf{P}$ (call it $\mathbf{P}_e$), as $\mathbf{P}_e = \mathbf{A}_e \cdot I\mathbf{P} + \mathbf{P}(0)$. Then iterations are used on $\mathbf{A}_e$ to minimize the least-squares error

$$E(\mathbf{A}_e) = \|\mathbf{P}_e(\mathbf{A}_e) - \mathbf{P}\|^2, \tag{16}$$

where $\|\cdot\|^2$ denotes the square of the norm; any reasonable norm can be used. One common norm is the Euclidean norm (see p. 98 in Ref. 15), this being the square root of the sum of the squares of the components of the vector for which the norm is being taken, that is, $\|x\|^2_{\text{Euclidean}} = x^{\text{transpose}} \cdot x$. However, in our case the Euclidean norm would vary with time and, thus, a theoretically more useful norm is the square root of the integral over the total time of the square of the Euclidean norm, that is, $\|x\|^2 = 1/T[\int_0^T \Sigma_{i=1}^n x_i(t)^2\, dt]$ when x is an n-vector. As a practical matter with experimental data, though, it is most convenient to replace the integral by a sum over the data points, that is, for $d\#$ the number of data points, to use

$$\|x\|^2 = \frac{1}{d\#} \sum_{k=1}^{d\#} \sum_{i=1}^{n} x_i(t_k)^2. \tag{17}$$

In using this least-squares technique, iterative changes are performed on the estimate $\mathbf{A}$ matrix, $\mathbf{A}_e$, using the gradient, ∇E, of $E(\mathbf{A}_e)$ with respect to $\mathbf{A}_e$. Writing the square matrix $\mathbf{A}_e$ as a vector of components A_{ei} we update the nth iteration of $\mathbf{A}_e$, $\mathbf{A}_e(n)$, to the $(n + 1)$th iteration by

$$A_e(n + 1)_i = A_e(n)_i - \lambda \cdot \nabla E_i, \tag{18}$$

in which λ is a free parameter used to insure convergence (normally $\lambda < 1$) and the gradient is defined as the vector of partial derivatives of E, of Eq.

(16), with respect to the corresponding components of the vector $\mathbf{A}_e$. In doing so, Eq. (18) essentially is the weight update formula for a neural network (18) in which case neural network theory can be used to an advantage in improving the performance [e.g., by adding momentum terms to Eq. (18)].

In any event the number of iterations needed becomes very large, generally in the 10,000 range, in order to achieve a reasonably small error, taking perhaps 15 min on a 50 MHz personal computer (PC). One can write a relatively simple program in something like QuickBASIC to find a suitable $\mathbf{A}_e$, or there are commercial programs for such purposes. Indeed we found that the program SCIENTIST (19) is reasonably efficient for carrying out this type of least-squares fit. Because it too uses least square fitting it may take some time. Other available programs include MatLab with its System Identification toolbox (20), though in practice it may be harder for nonengineers to use than SCIENTIST.

A very viable alternate is to use the precise form of the solution using convolution, as given in Eqs. (5)–(8). Because the convolution framework involves only $\mathbf{P}$ and $\mathbf{A}_e$, but not the integral $\int\mathbf{P}$ or the derivative of $\mathbf{P}$, it offers an improved means of proceeding numerically. As shown in the case of $N = 3$ above, by using Eqs. (7) and (8) and working down through the lower triangular state-space equations we can express the results in terms of the initial condition P_0 and the unknowns A_{rs}. In fact if the $P_{i,j}(t)$ are precisely known, these equations [Eqs. (9)–(11) for $N = 3$] can be solved precisely, showing that given N and $P_{i,j}(t)$ the K's are uniquely determined (this is also borne out by the fact that the exponentials are linearly independent and the convolution of functions on $[0,t]$ has no zero divisors). However, it should be pointed out that because of measurement noise and numerical error there could be a number of solutions for the K's that satisfactorily yield the proper cleavage results. On the other hand, by the example shown in Fig. 4, it also appears that in some cases small changes in the K's can yield significantly different cleavage pathways.

The convolution equations put us in a position to obtain an exact solution for $\mathbf{A}$, assuming exact data. As an example of the exact determination, from the first of Eq. (9)

$$\begin{aligned} P_0 &= P_{1,4}(0), \\ A_{11} &= \frac{\ln[P_{1,4}(t)] - \ln[P_0]}{t} \qquad \text{for any } t > 0. \end{aligned} \tag{19}$$

To obtain A_{21} and A_{22} we look at the derivative of $P_{1,3}$

$$\frac{dP_{1,3}}{dt} = \frac{A_{22}e^{A_{22}t} - A_{11}e^{A_{11}t}}{A_{22} - A_{11}} A_{21}P_0, \tag{20}$$

which, when evaluated at $t = 0$ yields A_{21} and then Eq. (9) gives A_{22}. For larger i,j we have to solve nonlinear (nondifferential) equations but these have unique solutions. However, experimental data are not exact and it is better to proceed by curve fitting the given data, $P_{i,j}(t)$, to the known form of the solutions, that is, to Eqs. (9)–(11) in the case of $N = 3$.

For the curve fitting procedure we note that the solutions are linear combinations of $\exp(A_{ii}t)$ where only those exponentials are present which go with rows at or above the $P_{i,j}$ being considered. Thus, we can use the diagonal entries of **A** to form an exponential vector

$$E\mathbf{P}(t) = [e^{A_{11}t}, e^{A_{22}t}, e^{A_{33}t}, \ldots,]^{\text{transpose}}. \tag{21}$$

Then the convolution solution evaluates to the form

$$\mathbf{P}(t) = \mathbf{B} \cdot E\mathbf{P}(t), \tag{22}$$

where **B** and $E\mathbf{P}(t)$ are functions of the entries of **A** with **B** being a lower triangular square matrix. Assuming distinct A_{ii}, for our $N = 3$ case we find **B** directly from Eqs. (9)–(11) to be 6×6 with the 36 entries

$$\begin{aligned}
&B_{11} = P_0; \quad B_{12} = B_{13} = B_{14} = B_{15} = B_{16} = 0,\\
&B_{21} = A_{21}P_0/(A_{11} - A_{22}) = -B_{22}; \quad B_{23} = B_{24} = B_{25} = B_{26} = 0,\\
&B_{31} = [A_{31}(A_{11} - A_{22} + A_{21}A_{32})]P_0/[(A_{11} - A_{22})(A_{11} - A_{33})],\\
&B_{32} = -A_{21}A_{32}P_0/[(A_{11} - A_{22})(A_{22} - A_{33})],\\
&B_{33} = [A_{31}(A_{22} - A_{33} - A_{21}A_{32})]P_0/[(A_{11} - A_{33})(A_{22} - A_{33})],\\
&B_{34} = B_{35} = B_{36} = 0,\\
&B_{41} = A_{41}P_0/(A_{11} - A_{44}) = -B_{44},\\
&B_{42} = B_{43} = B_{45} = B_{46} = 0,\\
&B_{51} = [A_{21}A_{52}(A_{11} - A_{44} + A_{41}A_{52})(A_{11} - A_{22})]P_0/\\
&\qquad [(A_{11} - A_{22})(A_{11} - A_{44})(A_{11} - A_{55})],\\
&B_{52} = -A_{21}A_{52}P_0/[(A_{11} - A_{22})(A_{44} - A_{55})],\\
&B_{54} = -A_{41}A_{54}P_0/[(A_{11} - A_{44})(A_{44} - A_{55}),\\
&B_{55} = [A_{21}A_{52}(A_{44} - A_{55} + A_{41}A_{54})(A_{22} - A_{55})]P_0/\\
&\qquad [(A_{11} - A_{55})(A_{22} - A_{55})(A_{44} - A_{55})],\\
&B_{53} = B_{56} = 0,\\
&B_{61} = [A_{61}(A_{11} - A_{44} + A_{41}A_{64})]P_0/[(A_{11} - A_{44})(A_{11} - A_{66})],\\
&B_{64} = -A_{41}A_{64}P_0/[(A_{11} - A_{44})(A_{44} - A_{66})],\\
&B_{66} = [A_{41}A_{64} - A_{61}(A_{44} - A_{66})]P_0/[(A_{11} - A_{66})(A_{44} - A_{66})],\\
&B_{62} = B_{63} = B_{65} = 0.
\end{aligned} \tag{23}$$

Although these equations look formidable for finding **A** in terms of **B**, once

the A_{ii} terms are known (from the exponentials in time) the equations in Eq. (23) are linear in the other A_{ij} and, so, **A** is readily obtained from **B** and with **A**, the K's, using the other linear transformation of Eq. (15). Specifically, assuming that the A_{ii} are found independently (as they will be via estimation of the exponentials) direct calculations on the above equations for **B** give

$$\begin{aligned}
A_{21} &= B_{21}(A_{11} - A_{22})/P_0, \\
A_{31} &= [B_{31}(A_{11} - A_{33}) + B_{32}(A_{22} - A_{33})]/P_0, \\
A_{32} &= -\frac{B_{32}}{B_{21}}(A_{22} - A_{33}), \\
A_{41} &= B_{41}(A_{11} - A_{44})/P_0, \\
A_{52} &= -\frac{B_{52}}{B_{21}}(A_{44} - A_{55}), \\
A_{54} &= -\frac{B_{54}}{B_{41}}(A_{44} - A_{55}), \\
A_{61} &= [B_{61}(A_{11} - A_{66}) + B_{64}(A_{44} - A_{66})]/P_0, \\
A_{64} &= -\frac{B_{64}}{B_{41}}(A_{44} - A_{66}).
\end{aligned} \tag{24}$$

The curve fitting procedure is then to evaluate Eq. (22) with an estimated **B**, call it $\mathbf{B}_e$, that is, $\mathbf{P}_e(t) = \mathbf{B}_e E\mathbf{P}(t)$, evaluate the norm, $\|\mathbf{P}_e - \mathbf{P}\|$, and update the **B** by using the gradient of the error, exactly as at Eq. (18) (with $\mathbf{A}_e$ replaced by $\mathbf{B}_e$). Once the error is small enough, $\mathbf{A}_e$ is found from $\mathbf{B}_e$ and from $\mathbf{A}_e$ the K's. One could also proceed directly to get the **A** matrix without the intermediary of the **B** matrix, but the gradient with respect to the entries of **A** appears to be numerically more noisy than that for the entries of **B** due to the convolution placing entries of **A** into denominator terms.

Because each row of **P** is independent of the lower rows and only introduces one new exponential, it is expedient to do a least-squares fit one row at a time. In that case the (transposed) parameter vector to be updated at each iteration becomes $[A_{ii}; B_{i1}, B_{i2}, \ldots, B_{ii}]$, and the gradient column vector, using the norm defined by Eq. (17), becomes

$$\nabla E_i = \frac{1}{d\#} \sum_{k=1}^{d\#} 2\{P_{ei}(t_k) - P(t_k)\}[t_k B_{ii} e^{A_{ii}t_k}; e^{A_{11}t_k}, e^{A_{22}t_k}, \ldots, e^{A_{ii}t_k}]^{\text{transpose}}, \tag{25}$$

where the subscript i on P_{ei} denotes the entry of $\mathbf{P}_e$ which goes with the ith row of **A**.

2. Transport of the granule to various parts of the cell for storage and release occurs (e.g., along an axon and into a terminal). In the model, the different physical regions are represented by multiple, diffusionally connected compartments (21–23). With different physiological situations, diffusional rates between compartments may conceivably vary.

3. Granules may be released from certain of the compartments at rates which depend on physiological conditions (21, 24).

4. Growth or change in the shape of the terminal must be taken into account in some cases, for example, in a growing organism (25, 26).

Movement of Granules within Terminal

Cell organelles, including secretory granules, are actively and directionally transported along microtubule tracts via specific attachment proteins (23, 27). Providing secretory granule concentrations are not high enough to be near saturating the microtubule transport system, it may be assumed that the migration of granules from one part of the cell (compartment) into another will occur at a rate proportional to the concentration of the granules in the compartment of origin. A first-order differential equation may thus be used to represent the process, so the secretory cell can be modeled as a set of compartments and connecting pathways using a multicompartmental anisotropic diffusion algorithm. The equations governing granule transport may then be written as (6):

$$dn_i/dt = \sum_{j \neq i} k_{ij} n_j / V_j - n_i \left(\sum_{j \neq i} k_{ji} + R_i \right) \Big/ V_i + S_i, \tag{26}$$

where n_i is the number of granules in the ith compartment, k_{ij} the rate constant for transport into ith compartment from jth (≥ 0), R_i the release rate constant from the ith compartment (≥ 0), S_i the rate of direct synthesis into ith compartment (≥ 0), and V_i the volume of the ith compartment.

Certain assumptions in the formulation in Eq. (26) must be noted, as they may be violated in real situations. First, granule concentration [n/V in Eq. (26)] is assumed to be without limits. Real granules are not point objects, and only a finite number can occupy a fixed volume. A corollary of this is that compartment volumes are assumed to be independent of granule content. Thus the standard diffusion model may not be valid at high concentrations. Second, granule transport and release rate constants are assumed to be independent of other model variables and time, including factors such as internal content, changing external molecular coats, second-messenger mod-

ulation of transport parameters, and size changes with age. The only form of time dependence we have included is for release probability to depend optionally on granule age.

If the coefficients in Eq. (26) are time independent, the equation can be written in matrix notation in a manner equivalent to Eq. (1):

$$d\mathbf{n}/dt = \mathbf{kn} + \mathbf{S}, \tag{27}$$

where bold letters indicate the column vector or square matrix equivalents of the terms in Eq. (26). In matrix notation the solution for the particular case of constant synthesis is

$$\mathbf{n} = e^{\mathbf{k}t} \cdot \mathbf{n}(0) + e^{\mathbf{k}t} * \mathbf{S}, \tag{28}$$

where $\mathbf{n}(0)$ is a column vector of initial conditions and $\mathbf{S}$ is a column vector of synthesis into each compartment. This gives for each compartment a time course of material concentration consisting of summed decaying exponentials no greater in number than the number of compartments [provided that all compartments have at least indirect access to release sites; Peikari (28)]. Each exponential in general differs in amplitude from compartment to compartment, but the rate constants of decay are shared by all compartments. Geometry can reduce the number of distinct exponentials in the solution. Because this case is too restrictive for many physiological situations of interest, we will leave this line of development and pursue a more general approach based on simulation.

Growth and Volume Changes

The granule transport model makes provision for a simple type of growth in which the geometry of the cell is assumed to expand uniformly over time. The rate constant for intercompartmental granule transport is assumed proportional to the area of the channel between the compartments. A doubling of the cross-sectional area produces a doubling of the transport apparatus and hence a doubling of the flux of material. Such an assumption implies that the rate-limiting step in transport is binding of vesicles to the transport system. The area grows as the square of the linear dimension and the volume as the cube. Thus, for a given number of granules in each compartment, transport and release rate constants are inversely proportional to the linear dimension:

$$dn_i/dt = \left[\sum_{j \neq i} k_{i,j} n_j / V_{oj} - n_i \left(\sum_{j \neq i} k_{j,i} + R_i\right) \Big/ V_{oi} + S_i\right] \Big/ v^{1/3}, \quad (29)$$

where v is the volume expansion factor giving the proportional increase in volume from a reference time, and V_{oi} and V_{oj} represent compartment volumes at the reference time. The latter serve to fix the geometrical ratios of volumes. The model also expands the synthetic apparatus of the cell in direct proportion to cell volume.

In the present configuration of the model, the volume factor, v, has a linear dependence (including one with 0 slope) on time, although other growth paradigms could be implemented easily. Thus

$$v = v_0 + k_g t, \quad (30)$$

where t is time, v_0 is the volume factor at $t = 0$, and k_g is a growth rate constant.

Age-Dependent Release

It has been observed in some secretory systems that newly synthesized granules are released preferentially. One possible mechanism would be for affinity of the release mechanism to decrease with granule age. We have included a simple age dependency as an option in our model. This modifies the probability that a granule, once targeted for release in the model, will actually be released. The probability decreases exponentially toward 0 starting from a value of 1.0 at an age of 0:

$$p_{rel} = e^{-k_{rel} t}, \quad (31)$$

where p_{rel} is the probability-modification factor and k_{rel} is the rate constant of probability decay.

Implementation of Model

The implementation of this model consists of two programs, named GRANULES and PROCON. GRANULES computes the age distributions of granules released from a model cell according to the schema described in the previous section. PROCON utilizes this age distribution to calculate the

mixture of peptide products according to the processing scheme described in the first section. Flow charts for the two programs are shown in Fig. 5.

In the GRANULES model (version 1.1; Fig. 5A), granules are represented by two one-dimensional arrays: one stores compartment-assignment integers for the granule (numbered consecutively from 1 to N_c), and the other the time of granule synthesis (birth date). Initializations include reading in an

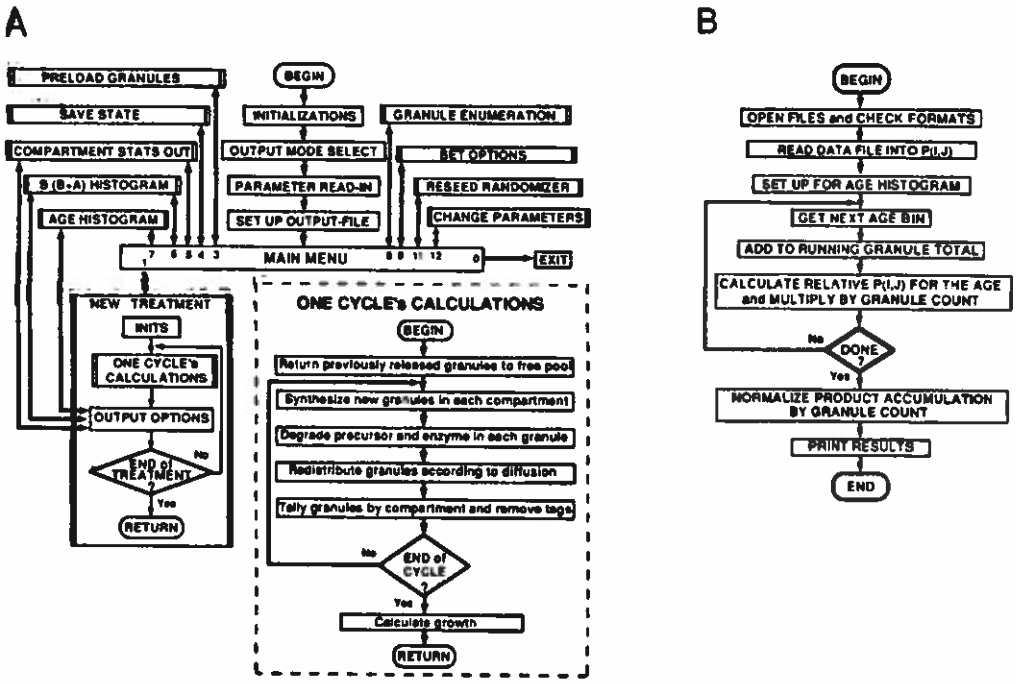

FIG. 5 Flow diagram for GRANULES model and peptide product calculations. (A) GRANULES. Program commences at BEGIN and follows the arrows. Various options are selected at the Main Menu, and then New Treatment is selected to initiate a model run. Calculations proceed by repeated calls to the subroutine ONE CYCLE's CALCULATIONS which performs simulations of granule synthesis and redistribution according to the compartmental and kinetic structure specified in an input-parameter file, as well as daily growth recalculations of model parameters. (B) PRO-CON program for computing peptide products from granule age distributions. The program reads a data file giving the time courses of occurrence of each of the peptide fragments in a granule, then convolves it with an age histogram file generated by GRANULES. This produces a net product distribution as a function of time for all peptides in a compartment or terminal (see text). Symbols are defined as follows: ovals, entry and exit points; double end-bar boxes, subroutine calls; diamonds, branch points or loops. [Modified from Hartline and Newcomb (13)].

ASCII format parameter file. The main menu gives options such as saving and restoring the state of the model on disk, changing parameters, and outputting results in various formats. When New Treatment is selected, the model is run for a user-specified period. A loop is entered to compute changes one cycle at a time (the time allotted to a cycle is arbitrary, being used only to determine the frequency of tallies and growth increments). Euler integration is applied in integrating first-order differential equations, meaning that time is divided into small (compared to the fastest kinetics) increments. At each time increment, the number of new granules to be added to each compartment is calculated from the compartment-specific synthesis rate multiplied by the volume factor. Empty slots are found in the granule arrays and filled with compartment assignment and birth date. Fractional granules for each compartment are held over and summed until a value greater than one accumulates, whereon an extra granule is synthesized for that compartment. Granules already in compartments are randomly selected, in numbers proportional to total compartment occupancy and intercompartmental diffusion constants, for transfer to connected compartments. Newly transferred granules are tagged and not allowed to move twice in the same time increment. A compartment with no back diffusion is used for granules released from the terminal. At the end of each time increment, tallies of granules in each compartment are made, and tags are removed. At daily intervals, ages of released granules are tallied, and histograms for granule age and extent of peptide processing (see next) are generated for outputting. Array-space for released granules is returned to the unassigned pool prior to initiation of the next days calculations.

To facilitate certain simple applications, our implementation of this model includes an option (available if fewer than 6300 granules are used) for two special granule arrays, one containing the remaining amount of prohormone in each granule and the other the rate constant of the processing enzymes. This may be used for simulation of one-cleavage-site processing, and includes the simulation of physiologically induced changes in both initial peptide amounts and processing activity in newly synthesized granules. For more complex situations, these quantities can be calculated from the granule age distributions.

The PROCON calculation of peptide fragment release concentrations as a function of time proceeds according to the flow diagram in Fig. 5B (PROCON version 1.0). After various initializations, a data file containing the stored time courses of peptide fragment kinetics is read into a fragment abundance array, computing normalized fragment distributions for each of the times corresponding to midpoints of the age histogram bins generated by GRANULES. Then the histogram of granule ages is read in one cycle at a time and one bin at a time. The number of granules in each bin is multiplied by

the fragment abundance for each fragment for that age. After the histogram of each time step has been processed, the number is normalized to (divided by) the total granule count for the histogram, stored on disk, and then the program goes to the next time step.

Model Applications

A single example is provided here of the combined operation of the GRANULES model and the peptide processing mathematics (Fig. 6). Additional examples may be found elsewhere (6, 13, 26). In the situation shown in Fig. 6, a prohormone containing three cleavage sites has fixed cleavage rate constants at each site, regardless of the length of the peptide containing the site (Fig. 6A). Cleavage at the middle site ($K_{i,j,2}$ = 0.02) is 15 times slower

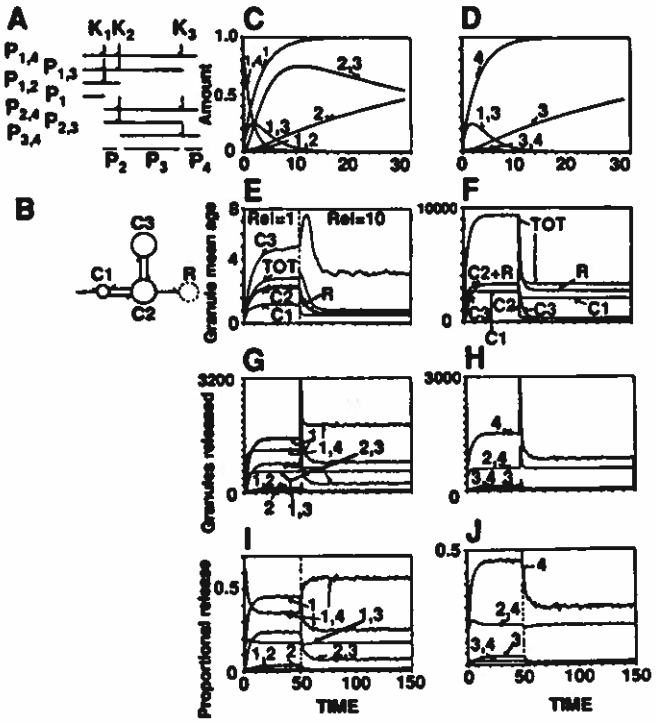

FIG. 6 Simulated output of a model peptidergic neuron using the GRANULES model. The model simulates a neuron with two compartments, one (upper in diagram) for granule storage; one for release.

than that at the two end sites ($K_{i,j,1} = K_{i,j,3} = 0.3$). The kinetic changes in the concentration of the various peptides in a granule as a function of time after synthesis are shown in Fig. 6C,D. Such granules are synthesized into compartment C1 of the cell, as diagrammed in Fig. 6B. The synthesis compartment communicates directly with a release compartment (C2), which in turn is connected to a storage compartment (C3). The rate constants for transport between all compartment pairs are equal and about the same as that for peptide degradation at the fast sites (=0.33). The model is started at $t = 0$ and run until a steady state in the granule content of the cell is reached, in which release balances synthesis (Fig. 6F). At $t = 10$ (time units are arbitrary), the release rate constant is increased 10-fold. Figure 6F shows the resulting sharp drop in the number of granules in each compartment (arrows labeled C1, C2, C3) and in total granule content of the cell. The rate of granule release (arrow R) increases sharply, then decreases as the granule supply in the cell is reduced (synthesis is held constant). Figure 6E shows the mean age of granules in each compartment under this protocol. Note that with the increase in release there is an initial increase in age of granules in the storage compartment (C3) as the supply of young granules in C2 is depleted. Figure 6G,H and 6I,J show the time courses for the amounts and proportions, respectively, of the different peptides released under the conditions modeled. The potential for physiologically significant effects not only from changes in amounts of bioactive peptides released (as would be expected with any transmitter system) but from changes in the proportions released, may be imagined.

Neuroendocrine Secretory Structures

The computer models presented here have been used to construct working hypotheses for secretory structures in which there are comparatively simple processing schemes [Table II; for details, see Newcomb *et al.* (6, 26)]. They have allowed us to make testable predictions of cellular or kinetic mechanisms.

First, the carboxyl-terminal truncation of the oxytocin neurophysin in the rat pituitary gland (29) has served as a model processing system in which experimental determination of *in vivo* processing rate constants under a variety of secretory conditions is feasible (26). The kinetic analysis of this allowed us to ask what actual *in vivo* processing rate constants are, at what rate they inactivate with time in secretory granules, and how the rate of propeptide conversion varies with changes in secretory activity. By showing a constancy of processing rate with changes in secretion rate, we can attribute changes in gland peptide content with physiological activity to changes in

TABLE II Summary of Quantitative Modeling of Peptide Processing in Neurosecretory Structures

Preparation	Measured quantities	Hypothesis tested	Ref.
Vertebrate neurohypophysis	Neurophysin conversion rate constants	Secretory granule turnover time and gland volume change with growth	Newcomb *et al.* (26)
	Changes in neurophysin content with age		
	Neurophysin conversion rate constants	Various hypotheses of secretory granule compartmentation and release	
	Changes in neurophysin content with *in vivo* depletion of gland content		
	Measurement of stored versus released neurophysin content *in vitro*		
Crustacean sinus gland	Stored peptide amounts in different animals	Biosynthetic mechanism (from stoichiometry)	Newcomb *et al.* (6)

secretory granule turnover times (as opposed to changes in processing enzyme activity) (26).

Analysis of gland oxytocin neurophysin content with growth and depletion of the neurohypophysis, in combination with various analyses of release and subcellular fractions, allows us to test the behavior of models for movement, storage, and release of secretory granules in the nerve endings of the posterior pituitary. In this approach, simulations of the peptide content of the neural lobe which include various rate constants for processing and transport kinetics, were compared with experimental measurements. Comparison of the model and experiments then suggested agreement, or specific experiments which will test the validity of models of cellular compartmentation (26).

Second, for the two-site internal cleavage of peptide H of the land crab sinus gland (8), modeling has shown how the approach can be used with a more complex prohormone in a case where the endocrine and cell biology of secretory structure is less completely understood. Application of the stoichiometric equations of processing allowed verification that a proposed biosynthetic mechanism does occur. Further use of the model showed how different cell biological situations can all give rise to experimentally observed

variations in sinus gland peptide content; experimental determination of the kinetics of propeptide conversion might be used to distinguish between these variations (6).

Future Directions

The mathematical and computer-simulation tools we have presented have so far been used only in relatively simple situations. Their potential power extends to substantially more complex situations. Such applications have been limited in the past by technical difficulties in biochemically monitoring concentrations of all relevant peptide products in a prohormone cleavage family. With improvement in biochemical techniques, more complex systems should be amendable to the type of analysis developed here.

As indicated in the development above, not all intermediates and final products need be measured in order to deduce the rate constants of cleavage. For simplicity, we have chosen to eliminate the final products from the calculations. Because these compounds are frequently the more accessible to measurement, it would be useful (and should be possible) to develop a formulation dropping some of the less accessible intermediates instead.

We have also indicated some of the technical difficulties with the numerical procedures we have applied. Finding an optimal procedure, especially one which is appropriate for noisy experimental data, is another challenge for the future.

The approach begun here can be extended to more complex situations still. Thus, issues dealing with the different effects of granule aging, targeting of granules to different release sites, and the modulatory control of many potentially susceptible sites in the synthesis, processing, transport and release system, can be addressed in a straightforward way in building more sophisticated models to further our understanding of endocrine systems. (Note: Copies of the simulation programs running on IBM PC-compatible microcomputers are available from the authors.)

Acknowledgments

Past work leading to this chapter has been supported by National Institues of Health Grants NS15314 to D.K.H., NS-24739 to R.Wh.N. and RCMI Grant RR-03061 to the University of Hawaii.

References

1. J. Douglass, O. Civelli, and E. Herbert, Polyprotein gene expression: Generation of diversity of neuroendocrine peptides. *Annu. Rev. Biochem.* **53**, 665–715 (1984).
2. T. Hokfelt, Neuropeptides in perspective: The last ten years. *Neuron* **7**, 867–879 (1991).
3. P. A. Rosa, P. Policastro, and E. Herbert, A cellular basis for the differences in the regulation of synthesis and secretion of ACTH/endorphin peptides in anterior and intermediate lobes of the pituitary. *J. Exp. Biol.* **89**, 215–237 (1980).
4. S. Zakarian and D. G. Smyth, Beta-endorphin is processed differently in specific regions of the rat pituitary and brain. *Nature (London)* **296**, 250–252 (1982).
5. C. Evans, D. L. Hammond, and R. C. Frederickson, The opioid peptides. *In* "The Opiate Receptor" (G. W. Pasternak, ed.), pp. 23–71. Humana Press, Clifton, New Jersey, 1988.
6. R. W. Newcomb, D. K. Hartline, and I. M. Cooke, Changes in information content with physiological history in secretory neurons. *Curr. Top. Neuroendocrinol.* **9**, 151–184 (1988).
7. B. I. Kanner and S. Schuldiner, Mechanism of transport and storage of neurotransmitters. *Crit. Rev. Biochem.* **22**, 1–38 (1987).
8. R. W. Newcomb, Amino acid sequences of neuropeptides in the sinus gland of the land crab *Cardisoma carnifex:* A novel neuropeptide proteolysis site. *J. Neurochem.* **49**, 574–583 (1987).
9. J. Bourdais, A. P. Pierotti, H. Boussetta, N. Barre, G. Devilliers, and P. Cohen, Isolation and functional properties of an arginine-selective endoprotease from rat intestinal mucosa. *J. Biol. Chem.* **266**, 23386–23391 (1991).
10. C. Evans, N. Maidmen, and R. Newcomb, Reversed phase liquid chromatography of biological peptides: Technical aspects and applications. IBRO Handbook Series, Vol. 15, "High Performance Liquid Chromatography in Neuroscience Research" (B. Holman, A. J. Cross, and M. H. Joseph, eds.), pp. 263–294. Wiley, New York, 1993.
11. K. G. Galanopoulou, S. N. Rabbani, N. G. Seidah, and Y. C. Patel, Heterologous processing of prosomatostatin in constitutive and regulated secretory pathways. *J. Biol. Chem.* **268**, 6041–6049 (1993).
12. C. S. Konkoy and T. P. Davis, Time-course administration of neuroleptics decreases regional neurotensin metabolism in intact brain slices. *Soc. Neurosci. Abstr.* **19**, 1364 (1993).
13. D. K. Hartline and R. W. Newcomb, Simulation of peptide processing, compartmentation and release in neurosecretory cells. *Neurochem. Int.* **19**, 281–296 (1991).
14. S. R. Searle, "Matrix Algebra for the Biological Sciences." Wiley, New York, 1966.
15. V. I. Arnold, "Ordinary Differential Equations." MIT Press, Cambridge, Massachusetts, 1978.
16. R. W. Newcomb, "Concepts of Linear Systems and Controls," Chapters 3 and 6. Brooks/Cole Publ., Belmont, California, 1968.
17. J. Mikusinski, "Operational Calculus," 2nd ed., Pergamon, New York, 1983.

18. S. Bhama and H. Singh, Single layer neural networks for linear system identification using gradient descent technique. *IEEE Transactions on Neural Networks* **4,** 884–888 (1993).
19. SCIENTIST, version 2.0, for DOS, MicroMath, Inc., Salt Lake City, Utah, 1993.
20. L. Ljung, "System Identification Toolbox for Use with MatLab." The Math Works, Natick, Massachusetts, 1993.
21. J. J. Nordmann, Hormone content and movement of neurosecretory granules in the rat neural lobe during and after dehydration. *Neuroendocrinology* **40,** 25–32 (1985).
22. J. F. Morris, D. B. Chapman, and H. W. Sokol, Anatomy and function of the classic vasopressin-secreting hypothalamus–neurohypophyseal system. *In* "Vasopressin" (D. B. Gash and G. J. Boer, eds.), pp. 1–90. Plenum, New York, 1987.
23. M. P. Scheetz, R. Vale, B. Schnapp, T. Schroer, and T. Reese, Movements of vesicles on microtubules. *Ann. N.Y. Acad. Sci.* **493,** 409–416 (1987).
24. J. J. Nordmann and G. Dayanithi, Release of neuropeptides does not only occur at nerve terminals. *Biosci. Rep.* **8,** 471–483 (1988).
25. G. I. Hatton, Cellular reorganization in neuroendocrine systems. *Curr. Top. Neuroendocrinol.* **9,** 1–28 (1988).
26. R. W. Newcomb, D. K. Hartline, J.-G. Lorentz, A. Depaulis, and J. J. Nordmann, Quantitative analysis and computer simulation of oxytocin–neurophysin processing in the rat neurohypophysis. *Neurochem. Int.* **19,** 297–312 (1991).
27. I. R. Gibbons, Dynein ATPases as microtubule motors. *J. Biol. Chem.* **263,** 15837–15840 (1988).
28. B. Peikari, "Fundamentals of Network Analysis and Synthesis." Prentice-Hall, Englewood Cliffs, New Jersey, 1974.
29. R. W. Newcomb and J. J. Nordmann, Quantative analysis of rat neurophysin processing. *Neurochem. Int.* **11,** 229–240 (1987).

Methods in Neurosciences

Volume 28

Quantitative Neuroendocrinology

Edited by

Michael L. Johnson

Departments of Pharmacology and Internal Medicine
University of Virginia Health Sciences Center
Charlottesville, Virginia

Johannes D. Veldhuis

Department of Internal Medicine
Division of Endocrinology and Metabolism
University of Virginia Health Sciences Center
Charlottesville, Virginia

ACADEMIC PRESS

San Diego New York Boston London Sydney Tokyo Toronto

1995