A Collection of 1,300 Dynamical Systems for Testing Data Fitting, Optimal Control, Experimental Design, Identification, Simulation or Similar Software

- User's Guide -

Address: Klaus Schittkowski
Department of Computer Science
University of Bayreuth
D - 95440 Bayreuth

E-mail: klaus.schittkowski@uni-bayreuth.de
Web: http://www.klaus-schittkowski.de

Abstract

We describe a collection of test problems which have been used to develop and test data fitting software for identifying parameters in explicit model functions, dynamical systems of equations, Laplace transformations, systems of ordinary differential equations, differential algebraic equations, or systems of one-dimensional time-dependent partial differential equations with or without algebraic equations. The test cases are useful also for any other reasons, e.g., in experimental design, identification, or optimal control. In addition, they might facilitate the implementation of new solvers for integrating ordinary or partial differential equations. All test problems and the corresponding data sets can be downloaded from the home page of the author. They were collected when implementing the interactive data fitting program **EASY-FIT**^{ModelDesign} , which is also available. All nonlinear model functions are implemented in a special modeling language called PCOMP providing function values and first and second partial derivatives. The corresponding Fortran codes of PCOMP are part of the software and data package.

1 Introduction

Parameter estimation plays an important role in natural science, engineering, and many other disciplines. The key idea is to estimate unknown parameters $p_1, \ldots, p_n$ of a mathematical model that describes a real life situation, by minimizing the distance of some known experimental data from theoretically predicted values of a model function at certain time values. Thus, also model parameters that cannot be measured directly, can be identified by a least squares fit and analyzed subsequently in a quantitative way.

In mathematical and somewhat simplified notation, we want to solve a least squares problem of the form

$$p \in I\!R^n : \quad \begin{array}{l} \min \sum_{i=1}^{l} \left(h(p, y(p, t_i), t_i) - y_i\right)^2 \\ p_l \leq p \leq p_u \quad , \end{array} \tag{1}$$

where $h(p, y, t)$ is a fitting function depending on the unknown parameter vector p, the time t, and the solution $y(p, t)$ of an underlying dynamical system. A typical dynamical system is given by differential equations that describe a time-dependent process, and that depend on the parameter vector p. Instead of minimizing the sum of squares, we may apply alternative residual norms, for example with the goal to minimize the sum of absolute residual values or the maximum of absolute residual values.

The mathematical background of the numerical algorithms is described in Schittkowski [285] in form of a comprehensive textbook. Also, the outcome of numerical comparative performance evaluations is found there, together with a chapter about numerical pitfalls, testing the validity of models, and a collection of twelve real-life case studies. Most of the case studies possess an industrial background.

The main goal of the documentation is to introduce a test bed designed for testing numerical algorithms for parameter estimation or data fitting, respectively, by a least squares fit. The mathematical model that is set up by a system analyst, has to belong to one of the following categories,

- explicit model functions,
- steady state systems,
- Laplace transforms of differential equations,
- ordinary differential equations,
- differential algebraic equations,
- one-dimensional time-dependent partial differential equations,
- one-dimensional partial differential algebraic equations.

However, the test examples can be used in many other situations as well, e.g., in

- dynamical systems,

- nonlinear optimization,
- system identification,
- optimal control,
- numerical solution of ordinary differential equations,
- numerical solution of partial differential equations,
- sensitivity analysis,
- automatic differentiation,
- Laplace transforms,
- experimental design.

All test problems are part of the database of **EASY-FIT**^{ModelDesign} , an interactive system for data fitting and experimental design in dynamical systems, which can be downloaded from the home page of the author. More information about **EASY-FIT**^{ModelDesign} is found in Schittkowski [285, 288]. All files containing the model functions in the PCOMP language, the data files for parameter estimation generated from the database of **EASY-FIT**^{ModelDesign} and the PCOMP source code files can also be downloaded through the link

`http://www.uni-bayreuth.de/departments/math/~kschittkowski/home.htm`

In Section 2, we outline briefly the mathematical structure of our main goal, to estimate parameters of dynamical systems. This helps to understand the data sets and the function files, especially for applications different from data fitting.

More detailed information about the models, especially in case of underlying ordinary or partial differential equations, is found in Section 3, where we also add lists of all test examples.

Model functions are implemented in a modeling language called PCOMP, see Dobmann, Liepelt, Schittkowski and Trassl [76], Dobmann, Liepelt, Schittkowski [75], Liepelt and Schittkowski [184], or Schittkowski [286]. The usage of the Fortran subroutines is documented in Section 4 together with an example, to parse the code, to evaluate function and derivative values, or to generate Fortran codes for function values and first derivatives.

In Section 5, we present detailed information how the input data files are organized, for example, how to retrieve experimental data for parameter estimation. The tables contain all relevant data by which the general model structure is defined, e.g., number of variables, differential equations, constraints, or fitting criteria.

Individual results obtained by data fitting, optimal control, or experimental design test runs are listed in Appendix A including number of function calls, number of iterations, objective function values, and execution times.

Appendix B contains a summary of the PCOMP modeling language and a list of error messages.

2 The Data Fitting Model

Primarily, the test problems have been collected to develop and test data fitting software, see Schittkowski [285, 288]. The goal is to estimate parameters in

- explicit model functions,
- Laplace transforms,
- steady state systems,
- systems of ordinary differential equations
- systems of differential algebraic equations,
- systems of one-dimensional, time-dependent partial differential equations,
- systems of one-dimensional partial differential algebraic equations.

Proceeding from given experimental data, i.e., observation times and measurements, the minimum least squares distance of measured data from a fitting criterion is to be computed that depends on the solution of the dynamical system.

The basic mathematical model is the least squares problem to minimize a sum of squares of nonlinear functions of the form

$$p \in I\!R^n : \quad \begin{array}{l} \min \sum_{i=1}^{l} f_i(p)^2 \\ p_l \le p \le p_u \end{array} \quad . \tag{2}$$

We assume that the parameter vector p is n-dimensional and that all nonlinear functions are continuously differentiable with respect to p. Upper and lower bounds are included to restrict the search area. $f_i(p)$ is a suitable fitting criterion which may depend on the solution of an underlying dynamical system, e.g., a system of ordinary differential equations.

However, our models a dynamic ones, i.e., depend on an additional parameter, in most cases the time. In addition, there might be an additional independent model parameter by which, e.g., a concentration or temperature value is to be specified from where a set of measurements is obtained.

To illustrate the situation, we omit possible multiple data sets, dependencies on underlying dynamical systems, and constraints on the parameters, and differ between three situations:

1. Time-dependent models: The model function $f_i(p)$ depends on the experimental time, i.e., we have measurements of the form

$$(t_i, y_i) \ , \ i = 1, \ldots, l \ \ , \tag{3}$$

moreover a model function $h(p,t)$, and we want to estimate the parameter vector p by minimizing

$$p \in I\!R^n : \quad \begin{array}{l} \min \sum_{i=1}^{l} (h(p,t_i) - y_i)^2 \\ p_l \le p \le p_u \end{array} \quad . \tag{4}$$

In this case, we define

$$f_i(p) = h(p,t_i) - y_i, \ i = 1, \ldots, l \ . \tag{5}$$

2. Time- and concentration dependent fitting criteria: The data fitting function $f_i(p)$ depends on the experimental time and an additional parameter which we call *concentration*. Any other physical meaning is, of course, allowed. We proceed from measurements of the form

$$(t_i, c_i, y_i) \ , \ i = 1, \ldots, l \ , \tag{6}$$

a model function $h(p,t,c)$, and we want to estimate the parameter vector p by minimizing

$$p \in I\!R^n : \quad \begin{array}{l} \min \sum_{i=1}^{l} (h(p,t_i,c_i) - y_i)^2 \\ p_l \le p \le p_u \end{array} \quad . \tag{7}$$

In this case, we define

$$f_i(p) = h(p,t_i,c_i) - y_i, \ i = 1, \ldots, l \ . \tag{8}$$

Advantages are the possibility to define a model as a function of t and c, and to generate three-dimensional plots. The drawback of this formulation, however, is that an underlying differential equation cannot depend on c as well, since we would have to evaluate the right-hand side of an equation also at intermediate times and would not know how to insert a suitable concentration value.

3. Time- and concentration dependent models: To overcome the drawback mentioned above, we assume that the dynamical model, say an ordinary differential equation, depends on an additional, in the statistical sense independent parameter c, i.e., c may be inserted into initial values, right-hand sides, fitting criterion, or even constraints. Now we proceed from measurements of the form

$$(t_i, c_j, y_{ij}) \ , \ i = 1, \ldots, l_t, \ j = 1, \ldots, l_c \ . \tag{9}$$

The model function is again given in the form $h(p,t,c)$, and we want to estimate the parameter vector p by minimizing

$$p \in I\!R^n : \quad \begin{array}{l} \min \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (h(p,t_i,c_j) - y_{ij})^2 \\ p_l \le p \le p_u \end{array} \quad . \tag{10}$$

Now we get the fitting criterion

$$f_{ij}(p) = h(p,t_i,c_j) - y_{ij}, \ i = 1, \ldots, l_t; \ , \ j = 1, \ldots, l_c \ . \tag{11}$$

In the subsequent sections, we proceed from the most general situation (10) and illustrate our approaches by examples based on dynamical systems like ordinary or partial differential equations.

3 Test Examples

3.1 Summary

The reason for attaching a comprehensive collection of test problems is to offer the possibility to try out discretization procedures or differential equation solvers, moreover data fitting, optimal control, experimental design, identification of similar software. The problems can be used for selecting a reference problem when trying to implement own dynamical models, or to test the accuracy and efficiency of numerical algorithms, for example in comparisons with other methods.

In most cases, parameter estimation problems are found in the literature or are based on cooperation with other academic or industrial institutions. In some other situations, however, differential equations are taken from research articles about numerical simulation algorithms, and are adapted to construct a suitable data fitting test problem. Thus, some model equations might not coincide exactly with those given in the corresponding references and the numerical solution is sometimes different from the one found in the reference.

We summarize a few characteristic data and the application background of the test problems that can be downloaded by the link

`http://www.uni-bayreuth.de/departments/math/~kschittkowski/home.htm`

Besides of problem name and some figures characterizing problem size, we present also information how measurement data are obtained,

E	-	experimental data from literature or private communication,
S0	-	simulation without error,
S05	-	simulation with uniformly distributed error of 0.5 %,
S1	-	simulation with uniformly distributed error of 1 %,
S5	-	simulation with uniformly distributed error of 5 %,
S10	-	simulation with uniformly distributed error of 10 %,
S50	-	simulation with uniformly distributed error of 50 %,
N1	-	simulation with normally distributed error, $\sigma = 1$,
N5	-	simulation with normally distributed error, $\sigma = 5$,
X	-	comparison with exact solution,
ED	-	experimental design,
none	-	no experimental data set, for example least squares test problem.

The difference between *simulated* and *experimental* data is that exact parameter values are known in the first case. Besides of a large collection of problems with practical experimental date, there are also a few others where the data are constructed, for instance in case of some optimal control problems, or are determined more or less *by hand*. In many other situations, the exact solution of the differential equation is known and used to simulate experimental data. These test examples can be used to check the accuracy of discretization formulae or ODE solvers.

Moreover, we show some references in the column headed by *ref*, from where further details can be retrieved. Either the data fitting problem is described in detail, or at

least the mathematical background of the model is outlined. In case of an empty entry, the model is provided by private communication and not published somewhere else, or a related reference is unknown to the author.

To summarize, we offer test problems for the following model classes:

explicit model functions	:	243
Laplace transforms	:	10
steady state equations	:	41
ordinary differential equations	:	575
differential algebraic equations	:	62
partial differential equations	:	325
partial differential algebraic equations	:	44
sum	:	1,300

3.2 Explicit Model Functions

We proceed from r measurement sets of the form

$$(t_i, c_j, y_{ij}^k)\ ,\ i = 1, \ldots, l_t,\ j = 1, \ldots, l_c,\ k = 1, \ldots, r$$

with l_t time values, l_c concentration values, and $l = l_t l_c r$ corresponding measured experimental data. Moreover, we assume that l weights w_{ij}^k are given. However, weights can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for functions for which experimental data do not exist. Thus, the subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation.

Usually, we proceed from the L_2- or Euclidean norm to formulate a parameter estimation problem of the form

$$p \in I\!R^n : \quad \begin{array}{ll} \min & \sum_{k=1}^{r} \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (w_{ij}^k (h_k(p, t_i, c_j) \ - \ y_{ij}^k))^2 \\ & g_j(p) = 0\ ,\ \ j = 1, ..., m_e\ \ , \\ & g_j(p) \geq 0\ ,\ \ j = m_e + 1, ..., m_r\ \ , \\ & p_l \leq p \leq p_u\ \ , \end{array}$$

where we assume that fitting criteria $h_k(p, t, c)$, $k = 1, \ldots, r$, and constraints $g_j(p)$, $j = 1, \ldots, m_r$, are continuously differentiable functions subject to p. The model function $h(p, t, c) = (h_1(p, t, c), \ldots, h_r(p, t, c))^T$ does not depend on the solution of an additional dynamical system and can be evaluated directly from a given parameter vector p that is to be estimated at given time and concentration values t and c. All explicit test problems are listed in Table 1.

Table 1. Explicit Model Function

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
2INDVARS	4	10	0	0	Problem with 2 independent model parameters		S5
2VALLEYS	2	4	0	0	Academic test problem with two local minima	[294]	E
3MODVARS	9	602	0	0	A demo problem with three independent model variables		E
4BAR_LNK	4	24	2	0	Design of a four bar linkage		X
ADI_CSTR	52	50	30	30	Steady-state adiabatic CSTR with irreversible first order reaction and errors in variables	[159]	E
APPRX3	6	10	0	0	Rational approximation of data	[21]	S5
ASPHER_1	6	53	2	0	Aspheric lens shape, lenticular measurements (1st data set)		E
ASPHER_2	6	27	2	0	Aspheric lens shape, lenticular measurements (2nd data set)		E
ASPHER_3	6	53	2	0	Aspheric lens shape, lenticular measurements (3rd data set)		E
ASPHER_4	6	53	2	0	Aspheric lens shape, lenticular measurements (4th data set)		E
ATROP_EX	4	24	0	0	Atropin-chase binding, linear model		E
BENNETT5	3	154	0	0	Superconductivity magnetization modeling (NIST study)		E
BIRDMILL	3	14	0	0	Non-identifiability	[294], [32]	S5
BOGGS2	3	3	1	1	Test problem of Boggs with rank deficient Jacobian at start and equality constraints	[282]	none
BOGGS8	5	3	2	2	Test problem of Boggs with rank deficient Jacobian at solution and equality constraints	[282]	none
BOXBOD	2	6	0	0	Biochemical oxygen demand (NIST study)	[45]	E
BURGER_W	3	50	0	0	Explicit solution of Burger's equation with eps=0.0005		N1
CAT_SEP	5	5	0	0	Catalysator separation problem		E
CEMENT	28	63	0	0	Hardening of cement	[68]	E
CENSUS	2	13	0	0	US census over years 1790 to 1900	[118]	E
CHEMOSTA	3	69	0	0	Steady-state chemostat	[80]	S5
CHWIRUT1	3	214	0	0	Ultrasonic reference block (NIST study)		E
CHWIRUT2	3	54	0	0	Ultrasonic reference block (NIST study)		E

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
COMBPROP	13	13	0	0	Combustion of propane	[138]	none
CYC_COM1	2	10	0	0	Stability of cyclodestrin complexes (MeBCd)		E
CYC_COM2	2	10	0	0	Stability of cyclodestrin complexes (HEtBCd)		E
DA_X	4	72	0	0	MDI simulation		E
DANWOOD	2	6	0	0	Energy radiated from a carbon filament lamp (NIST study)	[68]	E
DCA_CON	2	18	0	0	Decline curve analysis of reservoirs, constant percentage decline	[310]	E
DCA_HAR	2	18	0	0	Decline curve analysis of reservoirs, harmonic decline	[310]	E
DCA_HYP	2	18	0	0	Decline curve analysis of reservoirs, hyperbolic decline	[310]	E
DEGRAD3	3	20	0	0	Microbial degradation with hydrolysis and photolysis, explicit model		E
DEGRAD4	3	20	0	0	Microbial degradation with hydrolysis and photolysis, parallel version, explicit model		E
DENSITY	6	37	0	0	Density data of a liquid crystal		E
DFBLASER	4	4	0	0	Dynamic characteristics of a semiconductor DFB laser		none
DFE1	8	9	0	0	Explicit test function with local solutions, cycling model function etc.		S0
DFE2	7	15	1	1	Explicit test function with time-dependent model change		S5
DIFFREAX	6	292	0	0	Diffusion and reaction in solid phase, explicit formulation		E
DISS_TAB	2	4	0	0	Dissolution of tablets		E
DNS	3	30	1	0	Feulgen-hydrolysis of DNS, biochemical reaction	[255]	E
DOAS	21	100	0	0	Differential optical spectral absorption		S0
DRUG	2	14	1	0	Bolus intravenous injection of a drug	[118]	E
E_FIT	3	12	0	0	Rational-exponential data fitting		E
ECKERLE4	3	35	0	0	Circular interference transmittance (NIST study)		E
ELA_TUBX	3	40	0	0	Waves propagating in a liquid-filled elastic tube (KDVB equation)	[152], [341]	S5
ENSO	9	168	0	0	Atmospheric pressure differences (NIST study)	[153]	E
ENZREAC	4	13	0	0	Enzyme reaction, rational approximation		E

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
EW_WAVEX	2	48	0	0	Wave propagation in media with nonlinear steepening and dispersion	[130]	S5
EXP_FIT1	2	28	0	0	Exponential data fitting		S5
EXP_FIT2	7	33	0	0	Exponential data fitting, explicit solution of linear ODE		S0
EXP_FIT3	2	27	0	0	Exponential data fitting		E
EXP_FIT4	5	19	0	0	Exponential data fitting		E
EXP_FIT5	10	20	0	0	Exponential data fitting		E
EXP_FIT6	2	4	0	0	Exponential data fitting		E
EXP_FIT7	2	11	0	0	Explicit test function		S0
EXP_P1	2	3	0	0	Test example: trigonometric functions, overdetermined	[301]	none
EXP_P2	2	3	0	0	Test example: rational functions, overdetermined	[301]	none
EXP_P4	20	20	0	0	Test example: linear functions	[301]	none
EXP_P5	2	2	0	0	Test example: polynomial functions	[301]	none
EXP_P6	2	2	0	0	Test example: polynomial functions	[301]	none
EXP_P7	2	1	0	0	Test example: polynomial functions, underdetermined	[301]	none
EXP_SMPL	2	80	0	0	Single term exponential model, large errors in data		S50
EXP_TEST	4	200	0	0	Overlap of two exponential terms		S5
EXP_TST	4	20	0	0	Explicit test function, sum of two exponential terms		S0
EXP2TERM	5	20	0	0	Two-exponential model		E
F_DEHYDE	4	465	0	0	Colorimetric determination of formaldehyde	[240]	N5
GAMMAS	7	27	0	0	Analysis of a gamma spectrum		E
GAUSS	16	401	0	0	Distribution of points in Cartesian space fitted to linear combination of Gaussian functions		E
GAUSS_3D	6	29	0	0	Distribution of points in Cartesian space fitted to linear combination of Gaussian functions		E
GAUSS_AR	16	256	0	0	Fitting of data to eight gaussians		E
GAUSS1	8	250	0	0	Two well-separated Gaussians (NIST study)		E
GAUSS2	8	250	0	0	Two slightly-blended Gaussians (NIST study)		E
GAUSS3	8	250	0	0	Two strongly-blended Gaussians (NIST study)		E
GEAR	6	33	2	0	Gear with six parts		E
GEO_PROB	3	1	2	2	Maximum distance from origin to intersection of ellipsoid with hyperboloid	[200]	none

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
GLU_RATE	4	13	0	0	In-vivo glucose turnover rate		E
GO1	5	15	0	0	Test problem for global optimization	[253]	none
GO2	5	15	0	0	Test problem for global optimization	[253]	none
GO3	3	3	0	0	Test problem for global optimization (system of 3 equations)	[253]	none
GO4	3	1	3	3	Test problem for global optimization (system of 3 equations)	[253]	none
GO5	1	3	0	0	Test problem for global optimization (system of 3 equations)	[253]	none
GO6	1	1	1	1	Test problem for global optimization (system of 3 equations)	[253]	none
HAHN1	7	236	0	0	Thermal expansion of copper (NIST study)		E
HEAT_XX	2	99	0	0	Linear diffusion with constant parameters, exact solution		X
HEMISPH	3	100	0	0	Center of gravity of hemisphere		none
HEXA	5	149	0	0	Longitudinal sound velocity as function of propagation direction		E
HIMMELBD	2	2	0	0	System of two equation with local L2-solution		X
HYDENZYM	5	41	0	0	Hydrophobe enzymes and substrates		S0
HYP_ELA	8	231	17	0	Incompressible isotropic elastic strain-energy function		S1
ILL_COND	100	100	0	0	Ill-conditioned test function, many parameters		X
INFINITE	3	21	0	0	Infinitely many solutions		S0
INTEG_X	3	25	0	0	Population dynamics	[248]	S5
INTPOL	3	10	1	0	Interpolation routines, also non-continuous, non-smooth formulation		S0
IO3EXP	3	201	0	0	Input-output three-exponential model	[325]	N1
IRON_CC1	2	8	0	0	Iron carbochlorination		E
IRON_CCH	2	17	0	0	Iron carbochlorination	[263],[351]	E
ISOMER_X	5	40	0	0	Thermal isomerization of alpha-pinene to dipentene	[44], [294]	E
KIRBY2	5	151	0	0	Scanning electron microscope (NIST study)		E
KS_FUNC	3	402	0	0	Kreiselmeier-Steinhauser function		X
LANCZOS1	6	24	0	0	Exponential nonlinear regression (NIST study)	[171]	E
LANCZOS2	6	24	0	0	Exponential nonlinear regression (NIST study)	[171]	E
LANCZOS3	6	24	0	0	Exponential nonlinear regression (NIST study)	[171]	E

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
LIFE	2	16	0	0	Nonlinear regression		E
LIN_CMP1	7	10	3	2	Linear compartments with bolus administration, single dose	[135]	S5
LIN_CMP2	9	54	3	3	Linear compartments with multidose administration (extravascular)	[135]	S5
LIN_CMP3	3	19	0	0	Multidose administration (extravascular)	[135]	E
LIN_HC_X	3	165	0	0	Linear heat conduction	[2]	S5
LIN_KIN	6	32	0	0	Linear pharmacokinetic model with 3-compartments and lag time		S5
LIN_MOD	12	30	0	0	Linear data fitting with errors in time values	[330]	E
LIN_VIS	22	84	9	0	Linear-viscoelastic material law in frequency domain		X
LIQUID	9	34	1	1	Solution in soil		E
LKIN_X	3	28	0	0	Linear compartment model with bolus application and single dose		E
LKIN_X3	2	78	0	0	Linear compartment model with bolus application and three doses		S5
MAC_ECO	6	186	0	0	Macroeconomic time series of currency notes in circulation	[319]	N1
MARKET	7	100	0	0	Dynamic economic market		E
MARKET_C	7	750	0	0	Dynamic economic market, 750 data		E
MGH09	4	11	0	0	Rational nonlinear regression (NIST study)	[223]	E
MGH10	3	16	0	0	Exponential nonlinear regression (NIST study)	[223]	E
MGH17	5	33	0	0	Exponential nonlinear regression (NIST study)	[223], [241]	E
MICHMENT	2	12	0	0	Michaelis-Menten kinetics	[294], [356]	E
MIME	24	172	0	0	Revolution lines		E
MISRA1A	2	14	0	0	Monomolecular adsorption (NIST study)		E
MISRA1B	2	14	0	0	Monomolecular adsorption (NIST study)		E
MISRA1C	2	14	0	0	Monomolecular adsorption (NIST study)		E
MISRA1D	2	14	0	0	Monomolecular adsorption (NIST study)		E
MIX_PAT1	2	38	1	0	Mixing pattern inside a polymerization reactor		E

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
MIX_PAT2	3	33	0	0	Mixing pattern inside a polymerization reactor		E
MIX_PAT3	1	27	0	0	Mixing pattern inside a polymerization reactor		E
MIX_PAT4	3	28	0	0	Mixing pattern inside a polymerization reactor		E
MONOD	4	10	0	0	Monod-Wymnan-Changeux kinetic equation	[294], [261]	S5
MORTALTY	2	9	0	0	Mortality rate by Gompertz function	[140]	S5
NELSON	3	128	0	0	Analysis of performance degradation data (NIST study)	[233]	E
NLP_E5	4	41	0	0	Testing nonlinear programs with very many constraints	[283],[129]	none
NLP_L5	7	1001	8	2	Testing nonlinear programs with very many constraints	[283],[199]	none
NLP_P1	3	3	50	0	Testing nonlinear programs with very many constraints	[283][322]	none
NLP_P3	3	3	50	0	Testing nonlinear programs with very many constraints	[283],[322]	none
NLP_U3	5	251	0	0	Testing nonlinear programs with very many constraints	[283],[199]	X
OAT1	4	6	0	0	Bio-mass of oats	[267]	E
OAT2	3	6	0	0	Bio-mass of oats	[267]	E
OPT_KINX	6	60	2	0	Linear kinetics with variable switching times (optimal control problem)		none
OSCILL_S	16	50	0	0	Oscillating system with exact known solution	[369]	S0
OSCILL_X	16	50	0	0	Oscillating system	[369]	E
PARID120	3	121	0	0	Parameter identification model, 120 normally distributed experimental values		N1
PARID15	3	16	0	0	Parameter identification model, 15 normally distributed experimental values		N1
PARID30	3	31	0	0	Parameter identification model, 30 normally distributed experimental values		N1
PARID60	3	61	0	0	Parameter identification model, 60 normally distributed experimental values		N1
PERM1	4	4	0	0	Global optimization test problem (n=4, beta=50)		none
PERM2	4	4	0	0	Global optimization test problem (n=4, beta=0.5)		none
PERM3	10	10	0	0	Global optimization test problem (n=10, beta=1)		none

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
POL_APP	14	19	1	1	Polynomial approximation for computing axial forces		E
POL_MOD	14	30	0	0	Polynomial data fitting with errors in time values	[330]	E
POLARI	6	290	0	0	Fluorescence of polarization filter		E
POLYBU_X	5	68	0	0	Polymerization of high cis polybutadiene in hexane using catalyst (Euler scheme)		S5
PSS	5	5	1	0	Primary and secondary stable model		E
QUINIDIN	4	4	0	0	Population pharmacokinetics of quinidine	[69]	E
RAD_TRAC	3	17	0	0	Radioactive tracer in two human body compartments		E
RAMAN	2	303	0	0	Raman intensity of anisotrope probes	[172]	S5
RAT_APP	4	11	2	2	Rational approximation with constraints	[186]	E
RAT_FIT	4	11	0	0	Fitting a rational function	[164]	E
RAT42	3	9	0	0	Pasture yield with sigmoidal growth curve (NIST study)	[260]	E
RAT43	4	15	0	0	Dry weight of onion bulbs and tops (NIST study)	[260]	E
REFLECT	6	24	0	0	Reflection model for color design		E
REFLECTY	2	125	0	0	Reflectivity evaluation of electrons		E
RELEASE	4	32	0	0	Drug release		E
RICH_GR	3	9	0	0	Richards growth model	[266]	E
ROSZMAN1	4	25	0	0	Quantum defects in iodine atoms (NIST study)		E
RTD	2	26	1	0	Residence time distribution		E
SEQ_EXP	3	13	0	0	Sequential experiment	[294], [97]	E
SMOOTHNG	3	170	0	0	Data smoothing		E
SPLINE	2	6	0	0	Spline test		S0
STEP_RES	3	22	0	0	Second-order equation with dead time and step response data	[353]	E
SULFATE	4	17	0	0	Compartmental analysis in humans with radioactive sulfate as tracer	[294]	E
TEMP_LEV	6	92	0	0	Temperature level of goods		E
THERMRES	3	10	0	0	Thermistor resistance, exponential data fitting		E
THURBER	7	37	0	0	Semiconductor electron mobility (NIST study)		E
TIME_ACT	2	9	0	0	Time activities		E
TP1	2	2	0	0	Rosenbrock's banana function	[138]	none

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
TP1_A	2	2	0	0	Rosenbrock's banana function, ill-conditioned	[138]	none
TP1_B	2	2	0	0	Rosenbrock's banana function, very ill-conditioned	[138]	none
TP13	2	2	1	0	Academic test problem without constrained qualification	[138]	none
TP14	2	2	2	1	Constrained least squares problem	[138]	none
TP2	2	2	0	0	Constrained Rosenbrock's banana function	[138]	none
TP202	2	2	0	0	Academic test problem with attractive local solution	[282]	none
TP203	2	3	0	0	Simple data fitting problem	[282]	X
TP205	2	3	0	0	Least squares problem with three terms	[282]	none
TP212	2	2	0	0	Least squares problem with two terms	[282]	none
TP241	3	5	0	0	Least squares problem, five polynomial functions	[282]	none
TP242	3	10	0	0	Exponential test function	[282]	X
TP244	3	10	0	0	Exponential test function	[282]	X
TP246	3	3	0	0	Least squares problem with three terms	[282]	none
TP247	3	3	0	0	Least squares problem, helical valley in x3 direction	[282]	none
TP25	3	99	0	0	Academic test problem, highly unstable	[138]	X
TP256	4	4	0	0	Least squares problem with four terms, Powell's function	[282]	none
TP260	4	7	0	0	Least squares problem with seven terms	[282]	none
TP261	4	5	0	0	Least squares problem with exponential and trigonometric terms	[282]	none
TP267	5	11	0	0	Exponential test function	[282]	X
TP269	5	4	3	3	Constrained least squares problem with four linear terms	[282]	none
TP272	6	13	0	0	Exponential test function	[282]	X
TP282	10	11	0	0	Least squares problem with quadratic terms	[282]	none
TP286	20	20	0	0	Least squares problem with quadratic terms	[282]	none
TP288	20	20	0	0	Least squares problem, 20 linear terms	[282]	none
TP303	18	20	0	0	Least squares problem with squared sum	[282]	none
TP307	2	10	0	0	Exponential data fitting	[282]	E
TP308	2	3	0	0	Least squares problem with trigonometric terms	[282]	none
TP312	2	2	0	0	Least squares problem with two quadratic terms, local solution	[282]	none
TP327	2	44	1	0	Constrained exponential data fitting	[282]	E

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
TP332	2	201	2	0	Cam design problem	[282]	none
TP333	3	8	0	0	Exponential data fitting	[282]	E
TP334	3	15	0	0	Exponential data fitting	[282]	E
TP350	4	6	0	0	Rational approximation	[282]	E
TP351	4	7	0	0	Rational data fitting	[282]	E
TP352	4	40	0	0	Exponential and trigonometric data fitting	[282]	E
TP354	4	4	1	0	Constrained least squares problem, four quadratic terms	[282]	none
TP355	4	2	3	1	Constrained least squares problem, four quadratic terms and local solutions	[282]	none
TP358	5	20	0	0	Exponential data fitting test function	[282]	E
TP370	6	87	0	0	Complex least squares problem, six variables	[282], [241]	E
TP371	9	87	0	0	Complex least squares problem, nine variables	[282], [241]	E
TP372	9	6	12	0	Least squares problem, twelve inequality constraints	[282], [241]	none
TP373	9	6	6	6	Least squares problem, six equality constraints	[282], [241]	none
TP379	11	65	0	0	Test problem of Osborne, four exponential terms	[282], [241]	E
TP394	20	40	1	1	Least squares problem with one equality constraint	[282]	none
TP43	4	1	3	0	Rosen-Suzuki test problem	[138]	none
TP46	5	4	2	2	Equality constrained academic test problem	[138]	none
TP48	5	3	2	2	Equality constrained academic test problem	[138]	none
TP57	2	44	1	0	Constrained exponential fit	[138]	E
TP6	2	1	1	1	Rosenbrock's banana function, Betts' formulation	[138]	none
TP70	4	19	1	0	Chemical equilibrium problem	[138]	E
TPGLOB	3	3	0	0	Global solution of three equations in L1-form	[282]	none
TRAJ_NET	2	21	0	0	Test function with 5 local minima	[73]	E
TRANS90	3	22	0	0	Emittor at 90 deg		S5
TRANS90X	3	69	0	0	Emittor at 90 deg, experimental data		E
TREND	6	500	1	0	Trend curve		E
TRIG_APP	2	19	0	0	Trigonometric approximation for computing axial forces		E
TUBTANK	1	19	0	0	Comparison of tank and tubular reactors steady state	[147]	S5

(continued)

name	n	$\tilde{l}$	m_r	m_e	*background*	*ref*	*data*
VAPOR	2	11	0	0	Vapor-liquid equilibrium	[82]	E
VIS_SPEC	20	200	0	0	Viscoelastic spectra		N1
VISC_ELA	10	24	0	0	Memory function of visco-elastic substances		E
VISCREG1	3	23	0	0	Viscosity regression, data set 1		E
VISCREG2	3	103	0	0	Viscosity regression, data set 2		E
VISCREG3	3	1541	0	0	Viscosity regression, all data sets		E
WAVE_X	3	80	0	0	Explicit solution of wave equation		S5
WEIBULL	2	12	0	0	Weibull distribution		S5
ZS_EQTN	2	10	0	0	Zimm-Stockmayer equation	[95]	E

3.3 Laplace Transforms

Now we assume that the data fitting function is given in form of a vector-valued Laplace transform $H(p, s, c) \in \mathbb{R}^r$ depending on the parameter vector p to be fitted, the Laplace variable s, and an optional so-called concentration parameter c. Let function $h(p, t, c)$ be a numerical approximation of the inverse Laplace transform of $H(p, s, c)$, for instance computed by the formula of Stehfest [313], separately for each component.

Proceeding now from $l = l_t l_c r$ experimental data (t_i, c_j, y_{ij}^k) and weights w_{ij}^k, $i = 1, \ldots, l_t$, $j = 1, \ldots, l_c$, and $k = 1, \ldots, r$, we get the parameter estimation problem

$$p \in \mathbb{R}^n : \quad \begin{array}{ll} \min & \sum_{k=1}^{r} \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (w_{ij}^k (h_k(p, t_i, c_j) - y_{ij}^k))^2 \\ & p_l \le p \le p_u \end{array} .$$

General nonlinear constraints are not permitted in this case. Test problems defined by their Laplace transforms are listed in Table 2.

Table 2. Laplace Transforms

name	*n*	*l*	*background*	*ref*	*data*
CONCS	2	7	Test problem, only concentration values		S5
DIFFUS_L	1	99	Linear diffusion with constant parameters		S1
ELEC_NET	6	30	Electrical net		S5
LKIN_L	3	26	Simple linear compartment model, Laplace formulation		E
LKIN_L3	2	78	Simple linear compartment model, three initial doses		S5
PLASTER1	7	7	Pharmaceutic transdermal diffusion (plaster)	[362], [125]	E
PLASTER2	4	7	Pharmaceutic transdermal diffusion (plaster)	[362], [125]	E
PLASTER3	5	12	Plaster diffusion	[362], [125]	E
PLASTER4	2	12	Plaster diffusion	[362], [125]	E
TRAY	3	26	Marking tray hits		E

3.4 Steady State Equations

Again, it is supposed that r measurement sets of the form

$$(t_i, c_j, y_{ij}^k)\ ,\ i = 1, \ldots, l_t,\ j = 1, \ldots, l_c,\ k = 1, \ldots, r$$

are given with l_t time values, l_c concentration values, and $l = l_t l_c r$ corresponding measured experimental data. Moreover, we have weights w_{ij}^k. Weights can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for state variables for which experimental data do not exist. Thus, the subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation.

Together with an arbitrary fitting criterion $h(p, z, t, c)$, we get the parameter estimation problem

$$p \in I\!R^n : \quad \begin{array}{ll} \min & \sum_{k=1}^{r} \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (w_{ij}^k (h_k(p, z(p, t_i, c_j), t_i, c_j) \ - \ y_{ij}^k))^2 \\ & g_j(p) = 0\ ,\quad j = 1, ..., m_e\ , \\ & g_j(p) \geq 0\ ,\quad j = m_e + 1, ..., m_r\ \ , \\ & p_l \leq p \leq p_u\ \ . \end{array}$$

We assume that fitting criteria $h_k(p, z, t, c)$, $k = 1, \ldots, r$, state variable $z(p, t, c)$, and constraints $g_j(p)$, $j = 1, \ldots, m_r$, are continuously differentiable functions subject to p.

The state variable $z(p, t, c) \in I\!R^m$ is implicitly defined by the solution z of a system

$$\begin{array}{rcl} s_1(p, z, t, c) & = & 0\ , \\ & \cdots & \\ s_m(p, z, t, c) & = & 0 \end{array}$$

of nonlinear equations. All steady state test problems are listed in Table 3. Since none of them possesses additional constraints, the corresponding figures m_r and m_e are omitted.

Table 3. Steady State Equations

name	n	$\tilde{l}$	m	*background*	*ref*	*data*
ABSORP	1	10	7	Adsorption with surface complexation		E
BLOOD_S	3	32	1	Blood ethanol concentration	[342]	E
CENTRI	3	11	1	Ultracentrifuge for molecular weight determination		E
CHARGE	4	9	1	Charge relulation model, zero-potential as function of pH		S1

(continued)

name	n	$\tilde{l}$	m	*background*	*ref*	*data*
CHEM_EQU	2	20	4	Chemical equilibrium system		E
CHEMSTAT	2	10	3	Optimal residence time of a chemostat	[82]	S5
CO2_SOL1	2	14	10	VLE model for CO2 solubility in aqueous amine solutions		E
DEFORM	3	137	0	Deformation of visco-elestic material		E
DEWPOINT	1	21	3	Dew point temperature for isobutanol and water mixture	[300], [204]	S1
DISS_ENZ	2	30	1	Inhibition of dissociative enzymes	[166]	S5
MD_EQUI	1	6	1	Monomer-dimer equilibrium	[166]	S5
MDT_EQUI	2	6	1	Monomer-dimer-tetramer equilibrium	[166]	S5
METHANE	2	12	2	Partial oxidation of methane with oxygen	[123], [204]	S5
MULT_CST	4	20	8	Four stage CSTR battery in steady-state	[353]	S5
NA_CSTR	3	7	2	Continuous-flow stirred tank reactor (steady-state, normalized)	[343], [58]	S5
PERIA	5	30	2	Pulsar problem of astronomy		S1
RECLIC19	4	13	3	Receptor-ligand binding study	[270], [96]	S1
RECLIG1	2	33	2	Saturation curve 3H-compound on receptor membrane, one receptor and one ligand	[270]	E
RECLIG10	4	10	3	Receptor-ligand binding study	[270], [96]	S5
RECLIG11	2	34	2	Displacement curve with one receptor, one ligand	[270], [96]	E
RECLIG12	4	20	3	Displacement curve with one receptor, two ligands	[270], [96]	E
RECLIG13	4	22	3	Saturation curve	[270], [96]	E
RECLIG14	7	24	4	Displacement curve of quinpirole	[270], [96]	E
RECLIG15	7	16	4	Mass equilibrium model with two receptors and two ligands	[270], [96]	E
RECLIG16	4	20	3	Mass equilibrium model with two receptors and one ligand	[270], [96]	E
RECLIG17	2	7	2	Mass equilibrium model with one receptor and one ligand	[270], [96]	E
RECLIG18	4	14	3	Mass equilibrium model with one receptor, two ligands	[270], [96]	E
RECLIG2	3	27	3	Displacement curve of 3H-compound from one receptor, two ligands	[270], [96]	E
RECLIG3	4	10	3	Saturation curve, two receptors and one radioligand	[270], [96]	E
RECLIG4	6	75	4	Displacement curve of a 3H-compound with a substance, two receptors and ligands	[270], [96]	E
RECLIG5	4	11	3	Mass equilibrium model with one receptor and two ligands	[270], [96]	E
RECLIG6	3	12	3	Displacement curve	[270], [96]	E

(continued)

name	n	$\tilde{l}$	m	*background*	*ref*	*data*
RECLIG7	4	44	3	Displacement curve of 3H-compound from receptor	[270], [96]	E
RECLIG8	4	10	3	Saturation curve, two receptors and one radioligand	[270], [96]	E
RECLIG9	4	22	4	Displacement curve with cold ligand on receptor	[270], [96]	E
SING_EQU	2	7	1	Single equation		E
SS_REAC	4	30	6	Steady state reaction		S5
SULPHUR	4	90	3	Oxidation of sulphur dioxide to sulphur trioxid	[103], [204]	S1
TITRATIO	3	51	5	Potentiometric titration of N,N-dimethyhylaminoethylamine	[294]	S1
ULTRA1	3	6	1	Ultracentrifuge data analysis for molecular weight determination for one substance	[107]	S5
ULTRA2	3	6	1	Ultracentrifuge data analysis for molecular weight determination of two substances	[107]	S5

3.5 Ordinary Differential Equations

As before, we proceed from r data sets of the form

$$(t_i, c_j, y_{ij}^k) \ , \ i = 1, \ldots, l_t, \ j = 1, \ldots, l_c, \ k = 1, \ldots, r \ \ ,$$

where l_t time values, l_c concentration values and $l = l_t l_c r$ corresponding measurement values are given. Furthermore, we assume that l weights w_{ij}^k are defined. However, some of the weights w_{ij}^k can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for state variables for which experimental data do not exist. Thus, the subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation.

The data fitting function $h(p, y(p, t, c), t, c)$ depends on a concentration parameter c and in addition on the solution $y(p, t, c)$ of a system of m coupled ordinary differential equations with initial values

$$\begin{array}{rll} \dot{y}_1 = & F_1(p, y, t, c) & \ , \ y_1(0) = y_1^0(p, c) \ \ , \\ & \cdots & \\ \dot{y}_m = & F_m(p, y, t, c) & \ , \ y_m(0) = y_m^0(p, c) \ \ . \end{array}$$

Without loss of generality, we assume that, as in many real life situations, the initial time is zero. The initial values of the differential equation system $y_1^0(p, c)$, ..., $y_m^0(p, c)$ may depend on one or more of the system parameters to be estimated, and on the concentration parameter c.

The resulting parameter estimation problem can be written in the form

$$p \in I\!R^n : \quad \begin{array}{ll} \min & \sum_{k=1}^{r} \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (w_{ij}^k (h_k(p, y(p, t_i, c_j), t_i, c_j) \ - \ y_{ij}^k))^2 \\ & g_j(p) = 0 \ , \ \ j = 1, ..., m_e \ \ , \\ & g_j(p) \geq 0 \ , \ \ j = m_e + 1, ..., m_r \ \ , \\ & p_l \leq p \leq p_u \ \ . \end{array}$$

Again we have to assume that model functions $h_k(p, y, t, c)$ and $g_j(p)$ are continuously differentiable functions of p, $k = 1$, ..., r and $j = 1$, ..., m_r, and that the solution $y(p, t, c)$ is also a smooth function of p. All test problems based on ordinary differential equations are listed in Table 4. We do not list additional information about switching points or boundary values, for example.

Table 4. Ordinary Differential Equations

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
2BODY	2	80	4	0	0	Two-body problem	[144]	S5
2CSTR	3	80	4	0	0	Series of two CSTRs with time-delay	[239], [204]	S5
2LNK_ROB	2	40	4	0	0	Two-link planar robot without constraints	[8]	S5
2ND_ORD	3	10	2	0	0	Academic test problem, ill-behaved second order IVP	[48], [318]	S5
2ND_RATE	3	15	1	0	0	Second order rate equation under heat transfer conditions	[353]	S1
2STGCSTR	7	5	4	1	0	Time-optimal bang-bang control of two-stage CSTR	[204], [83]	none
ACTIVITY	2	9	2	0	0	Activities over time		E
ACTNITR	4	80	8	0	0	Nitrification in activated sludge process	[80]	S5
ADIABATI	2	30	2	0	0	Adiabatic complex gas-phase reaction in a PFR	[353]	S1
AEKIN	8	120	3	0	0	AE-kinetics		S1
AIRLIFT	1	87	2	0	0	Continuously stirred reactor with airlift		E
AIRY	2	38	2	0	0	Airy equation	[316]	S5
AKTIV_W2	8	128	4	0	0	Association kinetics, two-state-theory		S5
ALDRIN	2	10	1	0	0	Diffusion with chemical reaction		E
ALPHA_PI	5	52	5	0	0	Isomerization of an alpha-pinene		S5
AMENTO	5	48	12	0	0	Transport of Amentoflavone in Transwell plates		E
AMIDPRO	4	201	4	0	0	Amidproton replacement with protein folding		E
AMMONAB	3	39	3	0	0	Steady-state absorption column design	[147]	S1
AMYLASE	5	50	7	0	0	Alpha-amylase production with bacillus subtilis		S5
ANAEMEAS	5	72	7	0	0	Anaerobic reactor activity	[80]	S1
ANHYD	2	56	3	0	0	Oxidation of o-xylene to phthalic anhydride	[147]	S1
ANTIBIO	5	20	2	0	0	Kinetics of antibiotics in liquid manure	[266]	E
APHID	3	30	2	0	0	Number of aphids		E
APHIDEX	3	126	2	0	0	Number of aphids		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
APPRX1	5	20	1	0	0	Curve fitting	[360]	X
APPRX2	3	4	1	0	0	Curve fitting	[360]	E
ASS_CV1	11	57	7	0	0	Association curves		E
ASS_CV2	5	31	2	0	0	Association curves		E
ASS_CV3	6	27	2	0	0	Association curves		E
ASS_CV4	5	53	2	0	0	Association curves		E
ASS_CV5	7	47	3	0	0	Association curves		E
ASS_CV6	6	23	3	0	0	Association curves		E
ASS_CV7	7	23	2	0	0	Association curves		E
ASS_KIN1	3	15	1	0	0	Association kinetics		E
ASS_KIN2	4	15	1	0	0	Association kinetics with exponential term		E
ASS_KIN3	3	37	1	0	0	Association kinetics		E
ASS_KIN4	4	11	1	0	0	Association kinetics		E
ASS_KIN5	6	16	2	0	0	Association kinetics		E
ASS_KIN6	4	16	1	0	0	Association kinetics with exponential term and slope		E
ASTRO	1	80	4	0	0	Planar motion of earth around sun (singularities)	[8]	S5
ASYMP	3	27	2	0	0	Asymptotic boundary value problem	[8]	S5
AXDISP	3	80	16	0	0	Differential extraction column with axial dispersion	[147]	S5
B_BLOCK	10	41	2	0	0	Control of beta-blocker, two compartments	[60]	E
B_BLOCK1	20	41	2	0	0	Control of beta-blocker, two compartments	[60]	E
B_BLOCK2	40	41	2	0	0	Control of beta-blocker, two compartments	[60]	E
B_LYMPHO	8	12	3	0	0	B Lymphocytes in bone marrow	[116]	S5
BACTERIA	5	60	1	0	0	Bacteria population dynamics	[212]	S5
BALL	5	1	2	5	5	Bouncing ball	[303]	X
BARN1	3	22	2	0	0	Chemical reaction, Lotka-Volterra equation	[329]	E
BARN2	5	22	2	0	0	Chemical reaction, Lotka-Volterra equation with variable initial values	[329]	E
BATCH_CT	7	1	2	0	0	Control of nonlinear batch reactor	[202]	none
BATCHD	3	19	1	0	0	Dimensionless kinetics in a batch reactor	[147]	S5
BATCOM	8	364	4	0	0	Batch reactor with complex reaction sequence	[147]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
BATEX	2	20	2	0	0	Single solute batch extraction	[147]	S5
BATFERM	4	120	3	0	0	Batch fermentation	[80]	S5
BATSEG	2	10	2	0	0	Simple reaction with segregation in a batch reactor	[147]	S5
BATSEQ	4	44	4	0	0	Complex batch reaction sequence	[147]	S5
BCBPLUS	6	164	13	0	0	Reaction between brilliant cresol blue (BCB+) and acid chlorite		E
BEAD	3	90	6	0	0	Diffusion and reaction in a spherical bead	[147]	S5
BEER	17	62	7	0	0	Beer fermentation		E
BELLMAN	3	15	1	0	0	Chemical reaction (Bellman)	[342]	E
BELUSOV	3	132	4	0	0	Oscillating chemical reaction, highly stiff (Belusov-Zhabitinsky)		S5
BENZENE	2	16	2	0	0	Pyrolytic dehydrogenation of benzene to diphenyl		E
BENZHYD	2	20	2	0	0	Isothermal tubular reactor with two consecutive reactions (dehydrogenation of benzene)	[147]	S5
BI_OSC	2	100	2	0	0	Chaotic bi-stable oscillator	[43], [121]	S1
BIMOLECU	3	14	1	0	0	Carcino-embryonic antigen binding, bimolecular reversible reaction	[4]	E
BIO_MOD	2	300	3	0	0	Substrate production from biomass		S5
BIO_REAC	21	441	8	0	0	Biochemical dynamic model, highly overdetermined		E
BIODEG	8	42	3	0	0	Degradation of two substrates and growth of biomass		E
BIOKAT	5	100	3	0	0	Bimolekulare catalysis, Diels-Adler-reaction	[224]	S5
BIOMASS	2	10	2	0	0	Biomass and substrate of fermentor		S5
BIOPROC	4	15	3	0	0	Recombinant microbiological process	[84]	S5
BIOREAC	6	101	2	0	0	Biochemical reaction with enzyme deactivation	[12]	S1
BITUMEN	5	27	3	0	0	Modified Denbigh reaction scheme for converting bitumen into waste	[202], [71]	S5
BLASIUS	3	202	3	1	1	Incompressible laminar flow with zero pressure gradient	[143]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
BLOOD	10	124	9	0	0	Blood coagulation mechanism by thrombin formation	[319]	E
BLOOD_O	3	32	1	0	0	Blood ethanol concentration	[342]	E
BLYMPH	8	12	3	0	0	B-lymphocytes in a bone marrow	[116]	E
BLYMPH_R	3	12	3	0	0	B-lymphocytes in a bone marrow with reduced parameter set	[116]	E
BRUNHILD	6	21	3	0	0	Bolus injection of radioactive sulfate	[294]	E
BRUSSEL1	4	30	6	0	0	Multi-molecular reaction (Brusselator)	[181]	S5
BRUSSEL2	2	80	2	0	0	Multi-molecular reaction (Brusselator)	[126]	S5
BSTILL	4	180	11	50	0	Binary batch distillation column (nine floors)	[147]	S5
BSTILL_I	5	130	13	0	0	Binary batch distillation column (eleven floors)	[147]	S5
BVP	2	9	2	1	1	Boundary value problem	[8]	S5
BVP_NL	1	1	5	0	0	Nonlinear boundary value problem	[7]	none
BVP4	8	55	16	8	8	Complex 4-th order boundary value problem (normal mode decomposition of PDE)		S5
CABBAGE	8	24	3	0	0	Growth of white cabbage (roots, stem, leaves)	[266]	E
CAMY	6	568	2	0	0	Estimation of kinetic parameters of a chemical reactor		E
CAR_CTR1	21	1	2	2	2	Acceleration of a car		none
CAR_CTR2	22	1	2	3	2	Acceleration of car with possible break failure	[1]	none
CARGO	11	60	6	3	3	Transferring containers from ship to truck	[90], [323]	S1
CASC_IMP	2	10	11	0	0	Air humidity in laboratory device		S1
CASCADE1	15	9	5	1	1	Storage cascade of flow in pipes, Riccati equation	[185]	E
CASCADE2	3	9	1	0	0	Flow in pipes with one storage, Riccati-Muskingum equation	[185]	E
CASCSEQ	5	30	12	0	0	Cascade of three reactors with sequential reactions	[147]	S5
CASTOR	2	88	2	0	0	Batch decomposition of acetylated castor oil	[147]	S5
CAT_HYD	1	24	2	0	0	Catalytic hydrolysis of acetic anhydride	[343]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
CATALYST	10	1	7	0	0	Bifunctional catalyst blend of methylcyclopentane to benzene in a tubular reactor	[204]	none
CAV_BUBB	3	9	2	0	0	Cavitating bubble	[177]	S5
CH_CIRC	4	150	3	0	0	Electric circuit in a chaotic regime	[327]	S1
CHAIN_O1	4	40	2	0	0	First-order reversible chain reaction	[329]	S5
CHAN_FLO	3	23	4	2	2	Flow of a fluid during injection into a long channel	[78]	S1
CHANNEL	3	9	3	2	2	Flow in a channel (3rd order BVP)	[8]	S5
CHEM_OSC	10	50	5	0	0	Chemical oscillator	[139], [295]	S10
CHEM_REA	17	99	9	0	0	Chemical reaction		E
CHEMO	6	184	3	0	0	Chemostat fermentation	[80]	S5
CHSTA_D	8	54	5	1	1	Chemostat with dilution rate shift up, transient system		E
CIRCLE	2	402	4	0	0	Parameterized circle equation		S5
CIRCUIT	4	60	3	0	0	Electric circuit in a chaotic regime	[327]	N1
CLOUD	2	50	2	0	0	Behavior of spherical cloud of gas under gravitation	[70]	S5
COAL1	6	13	2	0	0	Coal pyrolysis, two parallel CH4 reactions	[52], [193]	S5
COAL2	11	86	3	0	0	Coal pyrolysis, concurrent reactions including CO, CO2, CH4, H2	[276], [193]	E
COAL3	3	20	1	0	0	Coal pyrolysis, first order H2 reaction	[276], [193]	E
COAL4	6	21	2	0	0	Coal pyrolysis, two parallel CO2 reactions	[276], [193]	E
COAL5	6	23	2	0	0	Coal pyrolysis, two parallel CO reactions	[276], [193]	E
COAL6	3	21	1	0	0	Coal pyrolysis, higher order CH4 reaction	[276], [193]	E
COAL7	18	86	6	0	0	Coal pyrolysis, parallel, higher reactions including CO, CO2, CH4, H2	[276], [193]	E
COLCON	4	50	11	0	0	Extraction cascade with backmixing and control	[147]	S5
COLLISIO	2	400	8	0	0	Collision dynamics between an Argon and a Neon atom in their mutual Lennard-Jones force field	[291]	S0

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
COMMENSA	3	18	7	0	0	Two bacteria with opposite substrate preferences	[80]	S5
COMP_ADS	10	40	8	2	2	Competitive adsorption of two components by multilayer model		E
COMP_EXP	4	38	2	0	0	Two compartments with equal absorption and exponential elimination	[254]	S5
COMPASM	3	46	5	0	0	Competitive assimilation and commensalism	[80]	S5
COMPET	4	50	2	0	0	Competition of two species	[43], [26]	S5
COMPREAC	8	154	7	0	0	Complex reaction scheme between formaldehyde and sodium para phenol sulphonate	[147]	S5
COMPSEG	2	60	6	0	0	Complex reaction with segregation in a semi-batch reactor	[147]	S5
CON_BURG	1	22	2	10	1	Burgers' equation with state and boundary constraints	[30]	S5
CONC4	7	35	1	0	0	Chemical simulation model		E
CONC4A	7	35	3	0	0	Chemical simulation model, alternative formulation		E
CONF_ALT	6	23	2	0	0	Conformation alterations of proteins		E
CONFLO1	2	40	1	0	0	Continuous open tank flow	[147]	S5
CONFLO2	2	40	1	0	0	Continuous closed isothermal tank flow	[147]	S5
CONFLO3	2	40	1	0	0	Continuous closed adiabatic tank flow	[147]	S5
CONINHIB	2	70	2	0	0	Continuous culture with inhibitory substrate	[80]	S5
CONSREA	2	16	12	0	0	Consecutive reactions with variable orders from combustion chemistry (optimized design)	[10]	S5
CONSTILL	6	60	10	0	0	Continuous binary distillation column	[147]	S5
CONTCON	3	44	3	0	0	Feed rate control of inhibitory substrate in a continuous culture	[80]	S5
CONTUN	2	105	4	0	0	Controller tuning problem	[147]	S1
CONVER	6	2	6	0	0	Control of monomer conversion and number average chain length		none
COOL	2	48	9	0	0	Continuous stirred-tank cascade	[147]	S5
COOL_CRI	2	374	5	0	0	Cooling crystallization (Miller and Parsival formulation)		E
COPPER	5	151	27	0	0	Multilayer model for moisture adsorbed on copper		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
COPPER_D	4	374	21	0	0	Multilayer model for moisture adsorbed on copper with desorption		E
COPPER2	5	277	7	0	0	Multilayer model for moisture adsorbed on copper		E
CR_ELOV	4	36	2	0	0	Chemical reaction		E
CRANE	5	18	6	0	0	Optimal control of a container crane	[274]	S5
CS_REAC	2	20	4	0	0	Continuously stirred reactor	[29]	S1
CST_1ORD	5	40	2	0	0	First order continuous stirred tank with cooling coil	[353]	S1
CSTOHNE	3	400	3	0	0	Compedition NH-replacement without reverse reactions		E
CSTR	2	60	3	0	0	Continuous stirred-tank cascade	[147]	S5
CSTR_BM	4	76	4	0	0	CSTR, benchmark example	[29]	S5
CSTR_CTL	20	100	2	0	0	Setpoint control of continuous stirred tank reactor	[99]	none
CSTR_CTR	7	1	3	0	0	Control of continuously stirred tank reactor	[175], [204]	none
CSTR_DFT	8	594	3	3	3	Capillary of Balzers		E
CSTR_JY0	5	306	4	0	0	Filter/capillary simulation		E
CSTRCOM	3	85	5	0	0	Isothermal reactor with complex reaction	[147]	S5
CUO	6	22	4	1	1	Multilayer model for moisture adsorbed on copper surface		E
DCIMMIGR	4	63	2	0	0	Linear model with time lags and immigration		E
DCKIN	2	20	1	0	0	Kinetic reaction with badly scaled exact solution		S1
DCMDEG	4	18	20	0	0	Dichloromethane in a biofilm fluidized sand bed	[80]	S5
DEACT	3	49	3	0	0	Deactivating catalyst in a CSTR	[147]	S5
DEACTENZ	3	90	7	0	0	Reactor cascade with deactivating enzyme	[80]	S5
DECAY	3	20	3	0	0	Radioactive decay of an isotope		S5
DEGEN	1	20	2	0	0	Notorious academic example, highly degenerate	[39]	S5
DEGEN_M	1	40	2	0	0	Modified notorious academic example, highly degenerate	[39], [368]	S5
DEGRAD1	3	30	3	0	0	Microbial degradation with hydrolysis and photolysis		E
DEGRAD2	4	30	3	0	0	Microbial degradation with hydrolysis and photolysis, parallel version		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
DEHYBENZ	2	16	2	0	0	Pyrolytic dehydrogenation of benzene to diphenyl		E
DIABETES	6	20	5	0	0	Diabetes management		E
DIAUXIA	5	100	5	0	0	Diauxic growth of a microbe		S5
DIFDIST	4	36	10	0	0	Multicomponent differential distillation	[147]	S1
DIMER	4	20	2	0	0	Pharmakokinetic model with two substances and one dimer complex		E
DIODE	2	18	2	0	0	Tunnel-diode oscillator	[148]	S5
DIS_KIN1	10	168	5	0	0	Displacement kinetics of a pharmaceutical experiment, data set 1		E
DIS_KIN2	10	168	5	0	0	Displacement kinetics of a pharmaceutical experiment, data set 2		E
DISLIQU	1	144	6	0	0	Distribution of substrates in a chemical reactor, liquid phase		S5
DISORDER	3	18	2	0	0	Treating manic-depressive disorder with Lithium carbonate	[303]	S5
DISPLMNT	8	32	3	0	0	Displacement curve		E
DISRET_O	2	128	16	0	0	Non-isothermal tubular reactor with axial dispersion	[147]	S5
DISSOC	8	94	1	0	0	Dissociation kinetics		E
DISTPAR1	17	78	2	0	0	Distributed parameter system (highly unstable and overdetermined)		S0
DISTPAR2	18	78	2	0	0	Distributed ODE parameter system, data from DISTPAR1		E
DLA1	6	242	2	0	0	Dissociation limited association kinetics, data set 1		E
DLA2	6	242	2	0	0	Dissociation limited association kinetics, data set 2		E
DLA3	6	242	2	0	0	Dissociation limited association kinetics, data set 3		E
DLA4	6	242	2	0	0	Dissociation limited association kinetics, data set 4		E
DLA5	6	242	2	0	0	Dissociation limited association kinetics, data set 5		E
DLA6	6	242	2	0	0	Dissociation limited association kinetics, data set 6		E
DLA7	6	242	2	0	0	Dissociation limited association kinetics, data set 7		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
DLA8	6	242	2	0	0	Dissociation limited association kinetics, data set 8		E
DMDS	8	66	4	0	0	Catalytic conversion of dimethyldisulfide		E
DMDS_A	8	88	4	0	0	Transient catalytic conversion of dimethyldisulfide		E
DRUG_SCH	3	32	3	0	0	Optimal drug scheduling for cancer chemotherapy	[55]	S5
DRUGDIS1	2	3	2	0	0	Time-optimal drug displacement, warfarin and phenylbutazone, one jump	[204], [213]	none
DRUGDIS2	4	3	2	2	0	Time-optimal drug displacement, warfarin and phenylbutazone, three jumps	[204], [213]	none
DRY_FRI1	1	40	4	0	0	Two-mass oscillator with dry friction between bodies (implicit switching)	[86]	S0
DRY_FRI2	3	40	4	1	0	Two-mass oscillator with dry friction between bodies, two variable switching times	[86]	S0
DRY_FRI3	5	40	4	3	0	Two-mass oscillator with dry friction between bodies, four variable switching times	[86]	S0
DRYER	5	63	4	0	0	Non-isothermic shaft dryer		S5
DUAL	3	48	3	0	0	Dual substrate limitation	[80]	S5
DUCT	3	10	1	0	0	Duct design problem (boundary value problem)	[42]	S5
DYNAMO	2	120	3	0	0	Chaotic behaviour of coupled dynamos	[43], [26]	S1
EAR2MAR1	10	1	3	3	3	Optimal control of earth-to-mars transfer	[47]	none
EAR2MAR2	51	1	3	3	3	Optimal control of earth-to-mars transfer	[47]	none
ENTERO	4	27	4	0	0	Linear pharmaco-kinetic model with lag-time		E
ENZCON	3	51	3	0	0	Continuous enzymatic reactor	[80]	S5
ENZSPLIT	3	10	2	1	1	Diffusion and reaction: split boundary solution	[147]	S1
ENZTUBE	2	10	1	0	0	Tubular enzyme reactor	[80]	S5
ENZYM	6	28	2	0	0	Enzyme effusion problem	[342]	E
EQBACK	3	50	10	0	0	Multistage extractor with backmixing	[147]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
EQEX	2	15	2	0	0	Simple equilibrium stage extractor	[147]	S5
EQMULTI	3	50	10	0	0	Continuous equilibrium multistage extraction	[147]	S5
ESTER_GA	3	14	1	0	0	Esterification		E
ETHANOL	7	100	4	0	0	Ethanol fed-batch fermentation by S. cervisiae	[106]	S5
ETHFERM	8	69	7	0	0	Ethanol fed batch diauxic fermentation	[80]	S5
ETHYL	4	-1	3	0	0	Ethyleen hydrogenation model (data sets 1 and 2)		E
ETHYL1	4	1526	3	0	0	Ethyleen hydrogenation model (data set 1)		E
ETHYL2	4	1421	3	0	0	Ethylen hydrogenation model (data set 2)		E
EX_BREAK	5	26	2	0	0	Linear compartment model with application of 2nd dose		S0
EXO_REAC	6	157	4	0	0	Exothermic reaction with lag time		E
EXOTHERM	2	100	2	0	0	Exothermic n-th order reaction in closed vessel (normalized)	[343]	S5
EXP_INC	2	60	3	0	0	Exponentially increasing solutions	[368], [7]	N1
EXP_SIN	2	7	1	0	0	Exponential-sinus function	[303]	X
EXP_SOL	2	23	2	0	0	Exponential solution	[316]	S5
FAST	4	28	2	0	0	Test problem, fast steady-state	[316], [170]	S5
FBR	3	45	8	0	0	Fluidized bed recycle reactor	[80]	S5
FC_EVAP	3	33	3	0	0	Forced-circulation evaporator	[234]	S5
FED_BAT	4	10	2	0	0	Optimal feeding strategy for monod-type models by fed-batch experiments	[225]	S5
FED_BATE	4	10	2	0	0	Optimal feeding strategy for monod-type models by fed-batch experiments, time-dependent feed	[225]	S5
FED10	4	80	8	0	0	Fed-batch reactor for protein production by recombinant bacteria	[179], [203]	S5
FEDBAT	4	180	4	0	0	Fed batch fermentation	[80]	S5
FEDBATCH	25	192	12	0	0	Fed batch fermentation process of streptomy-ces tendae		E
FERMENT	3	56	5	0	0	Batch fermentation		S5
FERMNT	5	126	9	0	0	Fermentation model with jump in input function		S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
FERMPROC	4	150	3	0	0	Fermentation process in bioreactor with jump in dillution rate		S5
FERMTEMP	4	100	5	0	0	Temperature control of fermentation	[80]	S1
FIN	2	8	2	0	0	Temperature in a long fin	[24]	E
FISH_POP	8	30	3	0	0	Fish population of lake Baikal		E
FLUID_CL	2	10	2	1	1	Fluid with immersed cooling coil (BVP)	[353]	S5
FLUOR	7	11	6	0	0	Fast fluorescence rate of photosynthesis	[11]	E
FLUORES	8	38	39	0	0	Fluorescence induction problem	[317]	E
FLUORESC	8	152	39	0	0	Fluorescence induction problem	[317]	E
FOLDING1	7	69	4	0	0	Unfolding and refolding of ribonuclease T1	[238], [215]	E
FOLDING2	6	72	4	0	0	Unfolding and refolding of ribonuclease T1	[238], [215]	E
FOLDING3	5	42	4	0	0	Unfolding and refolding of ribonuclease T1	[238], [215]	E
FOLDING4	4	38	3	0	0	Unfolding and refolding of ribonuclease T1	[238], [215]	E
FOLDING5	5	38	5	0	0	Unfolding and refolding of ribonuclease T1	[238], [215]	E
FOREST	5	40	2	0	0	Growth of forest	[43]	S5
FRACTAK	7	24	2	0	0	On-off-kinetics of fractakine binding		E
FUNGI	13	11	1	0	0	Spread of fungi in the root systems of growing plants	[272], [49]	E
FUNGI_I	3	11	1	0	0	Spread of fungi in the root systems of growing plants	[272], [49]	E
FUP_OSCI	2	200	40	0	0	Series of masses coupled by springs (Fermi-Ulam-Pasta oscillator)	[61], [98]	S5
GAS_ABS1	2	100	20	0	0	N-plate gas absorber with constant inlet feed stream, 20 plates	[204]	S5
GAS_ABS2	2	100	200	0	0	N-plate gas absorber with constant inlet feed stream, 200 plates	[204]	S5
GAS_OIL	3	40	2	0	0	Catalytic cracking of gas oil	[329]	S5
GASCLOUD	2	26	2	0	0	Thermal behavior of a spherical cloud of gas	[316], [306]	S0
GASLIQ1	2	20	6	0	0	Gas-liquid mixing and mass transfer in a stirred tank	[147]	S1
GASLIQ2	3	30	6	0	0	Gas-liquid mixing and mass transfer in a stirred tank	[147]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
GIBBFUJI	14	948	8	0	0	Modeling gibberella fujikuroi growth and GA3 production in solid state fermentation	[111]	E
GLIDER	4	72	4	1	0	Flight of glider with upwind	[349]	S1
GLOBCO2	5	161	7	0	0	Global CO2 model, exchange of energy, water, and carbon between continents and atmosphere	[297]	S5
GLUCOSE	9	40	3	0	0	Glucose reaction	[256]	S5
GLUCOSE1	4	27	2	0	0	Minimal model for glucose and insulin kinetics	[273]	E
GLUCOSE2	8	54	3	0	0	Minimal model for glucose and insulin kinetics	[273]	E
GOLF	2	24	6	0	0	Flight of a golf ball	[165]	S5
GREASE	3	13	2	0	0	Grease film of a fluid under high pressure	[165]	E
GROWTH_H	2	50	1	0	0	Logistic growth with stock dependent harvest	[43]	S5
GYROS	3	80	7	0	0	Idealized gyroscope in terms of quaternions (integral invariant)	[86]	S0
GYROSCOP	2	48	3	0	0	Heavy symmetric gyroscope	[165]	S5
HAMILTON	3	1	6	2	2	Hamiltonian system, two-point boundary system	[154]	E
HEATEX	3	200	24	0	0	Dynamics of a shell-and-tube heat exchanger	[147]	S1
HIGH_ORD	1	1	7	0	0	Ordinary differential equation of order 7		X
HIRES	11	32	8	0	0	Growth and differentiation of plant tissue at high levels of irradiance by light	[128]	S5
HIRES_PA	5	32	8	0	0	Growth and differentiation of plant tissue, after priority analysis	[128]	S5
HIRUDIN	9	39	5	0	0	Hirudin binding to thrombin and to a chemical mutant of thrombin	[167]	S5
HIV	4	40	4	0	0	HIV-AIDS epidemic evolution	[298]	S5
HMT	2	42	2	0	0	Semi-batch manufacture of hexamethylenetriamine	[147]	S5
HOLD	6	8	1	0	0	Ligament material properties with nonlinear springs and dashpots		E
HOLDUP	3	48	7	0	0	Transient holdup profiles in an agitated extractor	[147]	S1

(continued)

name	n	$\bar{l}$	m	m_r	m_e	*background*	*ref*	*data*
HOLE	3	38	1	0	0	Academic test example with hole	[303]	S5
HOMPOLY	2	21	3	0	0	Homogeneous free-radical polymerization	[147]	S1
HYDROL	2	6	2	0	0	Batch reactor hydrolysis of acetic anhydride	[147]	S5
HYDROXY	2	36	1	0	0	CSTR approximation to plug flow reactor		E
IDENT1	4	31	2	0	0	Structurally globally identifiable model	[354]	S0
IDENT2	4	11	1	0	0	Gas production by metal dissolution of Volmer-Heyrovski	[354]	S0
IMPULSE	3	20	2	0	0	Impulse of nerve potential	[299]	S5
INC_STIF	2	14	2	0	0	Class of test problems with increasing stiffness	[157]	S5
INHIB	3	39	4	0	0	Gas and liquid oxygen dynamics in a continuous fermenter	[80]	S5
INTERLEU	16	63	28	0	0	Interleukin-13 binding kinetics	[168]	E
IONTRAN2	2	297	1	0	0	Ion transport through membrane, logistic differential equation		E
IRB6400	9	6	6	0	0	Optimal control model for the industrial robot IRB6400	[134]	none
ISO_2PHA	3	40	4	0	0	Van-de-Vusse reaction in isotherm, ideally mixed CSTR with two phases		S5
ISO_BAT	4	15	4	0	0	Ideal isothermal batch reactor	[82]	S5
ISOMER	5	40	5	0	0	Thermal isomerization of alpha-pinene to dipentene	[329], [44], [294]	E
ISOTOP1	15	108	9	0	0	Isotope dilution with nine compartments		E
ISOTOP2	28	108	9	7	7	Isotope dilution with nine compartments		E
JFIT	7	24	1	0	0	Chemical reaction		E
JUXTA	4	10	4	0	0	Site juxtaposition model		E
KATALY1	13	49	9	0	0	Test reaction for catalysts		E
KATALY2	19	192	12	0	0	Test reaction for catalysts		E
KEPLER	2	48	4	0	0	Modified Kepler problem	[8], [275], [127]	S5
KIDNEY	4	200	5	0	0	Class of stiff test problems	[293]	S5
KIN_PRO	7	130	10	0	0	Kinetic chemical process		E
KINMOD	3	102	1	0	0	Reaction kinetics		S5
KLADYN	3	80	4	0	0	Dynamic model for KLa	[147]	S5
KNEE	1	9	1	0	0	Knee problem	[67]	S5
LANDING	3	202	6	0	0	Emergency landing of a hypersonic flight system		S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
LASER	3	36	6	0	0	Amplify electro-magnetic radiation by stimulated emission	[28]	S5
LEG_POL	2	18	2	0	0	Legendre polynomial of order 2	[316]	X
LEPS	3	600	6	0	0	LEPS-contour of molecule D-C-H	[291]	S5
LIN_SYS	1	210	15	0	0	System of linear ODEs	[278]	S5
LINEWEAV	2	15	1	0	0	Lineweaver-Burk plot	[80]	S5
LISA	5	6	7	0	0	Low thrust orbital transfer of a LISA spacecraft	[344]	E
LKIN	3	26	2	0	0	Simple linear compartment model		E
LKIN_A1	4	26	2	0	0	Simple linear compartment model, one parameter overdetermined		E
LKIN_A2	3	26	2	0	0	Simple linear compartment model, two parameters overdetermined		E
LKIN_A3	6	26	2	0	0	Simple linear compartment model, three parameters overdetermined		E
LKIN_A4	3	26	2	0	0	Simple linear compartment model, one nearly redundant parameter		E
LKIN_LA	3	34	2	0	0	Simple linear compartment model with variable lag time		S5
LKIN_NUM	3	26	8	0	0	Simple linear compartment model, explicit derivatives		E
LKIN_O3	2	78	2	0	0	Simple linear compartment model with three doses		S5
LKIN_RE	3	26	2	21	0	Simple linear compartment model, dynamic constraints		E
LKIN_S	3	26	8	0	0	Simple linear compartment model with sensitivity equations		E
LKIN_T	3	26	2	0	0	Simple linear compartment model (ODE), approximation error		E
LOC_EQUI	4	10	2	0	0	Location and continuation of equilibria		S1
LOG_GROW	2	50	1	0	0	Logistic growth with constant harvest	[43], [198]	S5
LORENZ	6	16	3	0	0	Lorenz equation	[194]	S5
LORENZ_S	3	240	3	0	0	Lorenz equation, highly oscillating	[194]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
LOT_VOL1	3	200	2	0	0	Lotka-Volterra differential equation	[141]	S5
LOT_VOL2	4	20	2	0	0	Lotka-Volterra differential equation	[329]	S5
LYMPHO	10	90	12	0	0	Immune lymphocyte circulation	[221]	S5
MARINE	16	160	8	0	0	Marine population	[78]	E
MCSTILL	3	100	20	0	0	Continuous multicomponent distillation column	[147]	S1
MCSTILLX	8	110	20	0	0	Continuous multicomponent distillation column, design of input feed	[147]	ED
MECH_SYS	6	230	4	0	0	Mechanical oscillating system with elasticity, slack, and damping		E
MEMINH	3	70	3	0	0	Cell retention membrane reactor	[80]	S5
MEMSEP	2	36	6	0	0	Gas separation by membrane permeation	[147]	S5
MENDES	36	1680	8	0	0	Biochemical dynamic model		S5
MET_SURF	6	46	2	0	0	Metalloid surface		E
METHAN	6	48	3	0	0	Conversion of methanol to various hydrocarbons	[209]	E
METHYL	2	30	2	0	0	Thermal explosion of methyl nitrate (normalized)	[343]	S5
MICGROW	2	20	3	0	0	Fed-batch bioreactor with one growing biomass	[17]	S5
MICGROWX	4	20	3	8	0	Fed-batch bioreactor with one growing biomass, experimental design	[17]	ED
MICGROWY	23	20	3	37	0	Fed-batch bioreactor with one growing biomass, experimental design with input function	[17]	ED
MICGROWZ	23	5	3	37	0	Fed-batch bioreactor, experimental design with input function and weights	[17]	ED
MILK1	5	45	3	0	0	Mastitis with diapedesis of neutrophil, three equations		E
MILK2	4	45	4	0	0	Mastitis with diapedesis of neutrophil, four equations		E
MINWORLD	4	117	3	0	0	Mini-world with population, consumption, and environmental pollution	[43], [216]	S5
MIX_RAT1	3	7	1	0	0	Mixed rate model, chemical reaction		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
MIX_RAT2	3	11	1	0	0	Mixed rate model, chemical reaction		E
MIX_RAT3	6	11	1	0	0	Mixed rate model, chemical reaction (cubic fit for Qd0)		E
MIXPOP	2	180	3	0	0	Predator-prey population dynamics	[80]	S1
MM_META1	2	80	4	0	0	Metabolic process in urine and plasma, Michaelis-Menten kinetics	[154]	S0
MM_META2	2	80	4	0	0	Metabolic process in urine and plasma, Michaelis-Menten kinetics	[154]	S0
MMKINET	4	22	3	0	0	Kinetics of enzyme action	[80]	S5
MN_CTRL	7	10	1	1	1	Minimum-norm optimal control problem		none
MOISTURE	4	6	3	0	0	Moisture of granulates		E
MOON	1	10	1	0	0	One-dimensional earth-moon-spaceship problem	[231]	S5
MOT_TSP	9	40	4	3	3	Optimal control of motorized traveling salesman problem	[274]	S1
MOTION	3	62	4	0	0	Motion of a car in a arena	[8]	S5
MUBATCH	4	30	8	0	0	Multicomponent batch distillation	[147]	S5
MULTILAY	13	23	7	1	1	Multilayer model for adsorption/desorption of molecules onto a metallic surface		E
MUSCLE	5	320	3	0	0	Ca2+ release in skeletal muscle cells	[141]	S5
MYL_ESTR	10	32	4	0	0	Methyl ester hydrogenation	[25], [204]	E
NC_RHS	9	56	3	0	0	Non-continuous right-hand side		E
NITRIF	4	45	4	0	0	Batch nitrification with oxygen transfer	[80]	S5
NITRO	3	60	2	0	0	Conversion of nitrobenzene to aniline	[147]	S1
NITROGEN	3	14	1	0	0	Reversible homogeneous gas-phase reaction of nitrogen oxide	[329]	E
NL_CSTR	10	160	8	0	0	Nonlinear isothermal CSTR for photochemical reaction	[41], [15]	S5
NL_ODE	3	20	3	0	0	Nonlinear ODE		S5
NLIN_VI1	29	248	10	0	0	Nonlinear-viscoelastic material law in frequency domain		E
NLIN_VI2	29	500	10	9	0	Nonlinear-viscoelastic material law in frequency domain with constraints		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
NON_DIFF	3	26	2	0	0	Simple linear compartment model, non-continuous RHS		S0
NON_ISO	2	20	2	0	0	Non-isothermal reactor with time-dependent reactant and temperature		S5
NON_KIN	6	30	2	0	0	Nonlinear pharmacokinetic reaction		S5
NOSTR	2	74	3	0	0	Non-ideal stirred-tank reactor	[147]	S5
NTA1	9	38	3	0	0	C. heintzii grown on glucose switched to nitrilo-triacetic acid		E
NTA2	10	38	3	0	0	C. heintzii grown on glucose switched to nitrilo-triacetic acid		E
NUTRITI	4	40	4	0	0	Nutritive cycle with two competing plant populations	[43]	S1
OBSERV1	2	11	2	0	0	Linear observer in normal form		S5
OBSERV2	2	48	4	0	0	Linear observer		S5
OBST_CTL	29	58	3	29	0	Obstacle problem (optimal control)	[292]	X
OC_EX3	6	30	2	0	0	Optimal control test problem	[90], [230]	none
OC_EX4	1	50	2	0	0	Optimal control test problem (bang-bang solution)	[90], [230]	none
OEKOSYS	8	60	3	0	0	Ecological system with two trophic layers	[192]	S5
OIL	4	40	6	0	0	Oil shale pyrolysis	[359], [204]	S1
OLIGO	4	84	4	0	0	Oligosaccharide production in enzymatic lactose hydrolysis	[80]	S5
ON_OFF1	7	8	2	1	0	On-off kinetics with two lag times		E
ON_OFF2	8	31	2	0	0	On-off kinetics		E
ON_OFF3	10	35	3	0	0	On-off kinetics binding atropin-chase		E
ON_OFF4	10	67	4	0	0	On-off kinetics binding atropin-chase		E
ON_OFF5	6	35	2	0	0	On-off kinetics binding atropin-chase		E
ON_OFF6	10	27	3	0	0	On-off kinetics binding atropin-chase		E
ON_OFF7	8	30	2	0	0	On-off kinetics binding atropin-chase		E
OPT_CON	50	1	5	0	0	Yeo's optimal control problem	[205]	none
OPT_CONT	4	29	3	2	0	Optimal control problem with 2nd order state constraints	[349], [46]	S5
OPT_CTRL	21	19	5	4	4	Optimal control with four final states	[59], [202]	none

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
OPT_KIN	6	60	2	2	0	Optimal adoption of initial infusion and doses at given therapeutic level		E
ORB_MOTN	3	40	4	0	0	Simple orbit motion	[278]	S5
ORBIT	9	3	3	0	0	Minimum time orbit transfer (optimal control)	[90]	none
OREGO	3	24	3	0	0	Belusov-Zhabotinskii reaction (oregonator)	[128], [93]	E
OSC_REAC	3	25	3	0	0	Chemical oscillation		S5
OSC_TRAN	5	86	2	0	0	Oscillating system with transient influences		E
OSC2INTM	3	364	4	0	0	Oscillation of the concentration of two intermediates	[119]	S5
OSCIL	3	150	3	0	0	Oscillating tank reactor behavior	[147]	S5
OSCILL	5	115	6	0	0	Oscillations in a heterogeneous catalytic reaction		S1
OXDYN	2	29	3	0	0	Oxygen uptake and aeration dynamics	[80]	S5
OXENZ	2	24	3	0	0	Aeration of a tank reactor for enzymatic oxidation	[80]	S1
OXIDAT	3	40	3	0	0	Oxidation reaction in an aerated tank	[147]	S5
OZONE	2	30	2	0	0	Ozon kinetics in atmosphere	[303]	S5
PAR_SIZE	4	28	30	0	0	Population balance including coalescence and breakup rates	[311]	E
PARTICLE	2	2	4	2	2	Particle diffusion and reaction (2nd order BVP)	[8]	X
PCB	17	90	9	0	0	Kinetic analysis of catalytic hydrogenization of PCB		E
PEAKS	3	102	3	0	0	Stiff ODE with sharp peaks	[117]	S5
PECAN	4	9	2	0	0	Population size of pecan aphids		E
PEND_ELA	1	20	4	0	0	Elastic pendulum	[196], [86]	S0
PENDU_I0	2	80	5	0	0	Plain pendulum, index-0-formulation		S1
PENICILL	4	40	4	0	0	Fed-batch fermentor for biosynthesis of penicillin	[203], [66]	S5
PESTICID	4	20	4	0	0	Pesticide degradation with explicit microbial population dynamics	[265], [244]	E
PHA_DYN1	9	35	3	0	0	Pharmaco-dynamic model with variable individual initial lag-times		E
PHA_DYN2	19	35	3	0	0	Pharmaco-dynamic model with variable initial values, initial time 0		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
PHA_DYN3	4	35	3	0	0	Pharmaco-dynamic model with one fixed initial time for all experiments		E
PHA_DYN4	5	35	3	0	0	Pharmaco-dynamic model with one variable initial lag-time		E
PHA_DYN5	7	61	3	0	0	Pharmaco-dynamic reaction with lag time, I0 = 0.01		E
PHA_DYN6	7	61	3	0	0	Pharmaco-dynamic reaction with lag time, I0 = 0.1		E
PHA_DYN7	7	61	3	0	0	Pharmaco-dynamic reaction with lag time, I0 = 1		E
PHA_KIN1	6	11	3	3	2	Linear pharmaco-kinetic model with bolus administration	[135]	S5
PHA_KIN2	9	52	7	1	1	Linear pharmaco-kinetic model with 3 segment absorbations, single dose	[135]	S5
PHA_REAC	5	180	2	0	0	Pharmaco-dynamic reaction		E
PHARM_AP	2	10	1	0	0	Pharmaceutical application in spherical coordinates		S1
PHARMA	9	52	5	0	0	Linear compartmental pharmacological model		E
PHB	4	60	3	0	0	Structured model for PHB production	[80]	S5
PHOS_TRA	26	58	8	0	0	Reversible reactions of phosphotransfer system		E
PHOSPH_D	2	75	3	0	0	Chemical reaction, phosphorescence		S5
PHOTO	11	38	39	0	0	Photosynthesis process	[271], [317]	E
PHOTO_PR	2	24	1	0	0	Daily photoproduction of plants	[43]	S5
PHOTO_S	11	31	39	0	0	Photosynthesis process	[271], [317]	S0
PHOTOCON	7	705	3	0	0	Fast transient photoconductivity		E
PLANT_GR	8	36	2	0	0	Plant growth (reset of initial values)		S5
PLASMID	2	200	5	0	0	Stability of recombinant microorganisms	[80]	S5
PLATINUM	6	120	3	0	0	CO oxidation on platinum	[50]	S1
PLUG_FLO	3	20	2	0	0	Plug-flow tubular reactor	[55]	S1
POLLUTNT	5	46	2	0	0	Treatment of pollutant		E
POLY1	14	17	5	0	0	Polymerization		E
POLY2	8	13	4	0	0	Polymerization		E
POLYBU	5	68	5	0	0	Polymerization of high cis polybutadiene in hexane using catalyst		S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
POLYMER	3	7	2	0	0	Polymerization		E
POPUL	10	83	10	1	1	Population counts		S5
PROTEIN	4	60	5	0	0	Production of secreted heterologous protein in fed-batch reactor by a yeast strain	[247], [15]	S5
PROTOZOA	3	21	1	0	0	Logistic growth model of protozoa		S5
PYRIDIN	10	66	7	0	0	Denitrogenization of pyridin	[38]	E
PYROLYS	2	101	6	0	0	Pyrolysis of chemically treated biomass, thermogravimetry analysis of a sample		S5
RABBIT	3	80	2	0	0	Rabbits eat grass on an island		S5
RAMP	5	62	1	0	0	Ligament material properties with nonlinear springs and dashpots		E
RATE_MOD	6	36	3	0	0	Catalytic hydrodesulfurization of sulfur molecules (DBT)		E
RATSOL1	3	20	2	0	0	Existence of rational solution	[113]	S5
RATSOL2	2	20	2	0	0	Existence of rational solution	[113]	S5
RC_CAT	3	83	1	0	0	Rate constant of catalyst C4H10O and water		E
RD_CSTR	5	1194	6	1	0	Release distribution in liquid phase of CSTR with input control		E
RE_ENTRY	7	6	6	0	0	Apollo re-entry problem	[318]	E
REAC	3	40	10	0	0	Chemical reaction		S5
REAC_CTR	7	1	2	20	0	Control of first-order reversible chemical reaction with dynamic constraints	[162], [202]	none
REACMECH	5	64	5	0	0	Reaction mechanism with stiff differential equations	[291]	S5
REACTION	6	12	5	0	0	Chemical reaction		E
REFRIG	2	10	2	0	0	Auto-refrigerated reactor	[147]	S5
REG_RES	4	100	2	0	0	Dynamics of a population depending on a regenerative resource	[43], [114]	S5
RELAY	2	50	2	0	0	Simple discontinuous model with a relay	[304]	S1
REPFED	5	40	3	0	0	Repeated fed batch culture	[80]	S5
REPLCUL	5	14	2	0	0	Repeated medium replacement culture	[80]	S5
RES_TIME	1	1	4	0	0	Optimal residence time for maximum yield in an ideal isothermal batch reactor	[82]	E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
RESPIRA	6	101	2	0	0	Viscoelastic model of respiratory mechanics	[325]	N1
REVTEMP	2	48	4	0	0	Reversible reaction with variable heat capacities	[147]	S5
REXT	2	82	5	0	0	Reaction with integrated extraction of inhibitory product	[147]	S1
ROB_ARM	3	20	4	0	0	Robot arm with two links	[237]	E
ROB_CTRL	5	3	4	5	2	Time-optimal control of two-link robotic arm	[357], [204]	none
ROBERT	3	33	3	0	0	Robertson's differential equation for reaction rates	[269]	S5
ROBOT	8	102	6	0	0	Move equations of a robot (Manutec r3)	[243]	S5
ROBOT_2	17	6	4	4	0	Optimal control of robot with two arms	[351]	none
RODC	2	10	2	0	0	Radiation from metal rod	[147]	S5
ROESSLER	3	21	3	0	0	Roessler differential equation		S5
RT_PULSE	7	1	3	0	0	Rectangular pulse in right-hand side	[326], [204]	none
RUN	1	24	4	0	0	Relief on a runaway polymerization reaction	[147]	S1
SAT_EXP	6	33	3	0	0	Saturation experiment in pharmaceutics		E
SBR_COOL	5	40	7	0	0	Optimal control of semi-batch reactor with cooling	[1]	none
SCP	2	1	5	4	0	Singular control problem	[350]	none
SE	2	29	1	0	0	Single chemical reaction		E
SEIR	5	123	6	1	1	SEIR epedemic modl applied to SARS		S5
SEMIPAR	2	15	5	0	0	Parallel reactions in a semi-continuous reactor	[147]	S5
SEMISEG	3	30	3	0	0	Simple reaction with segregation in a semi-batch reactor	[147]	S5
SEMISEQ	2	30	5	0	0	Sequential reactions in a semi-continuous reactor	[147]	S5
SENS	6	51	3	0	0	Stiff academic test problem for testing sensitivity analysis	[155]	S5
SHAKER	5	42	2	0	0	Shaker table driven by DC mortor and transmission	[183]	S5
SHARP1	1	48	2	0	0	Sharp fronts	[56]	S1
SHARP2	1	40	2	0	0	Sharp fronts	[56]	S5
SHEEP	6	138	9	0	0	Transport of radiocaesium in sheep	[108]	S5
SHELL	6	44	9	0	0	Chemical reaction of aromates	[38]	E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
SIMP_ECO	4	20	2	0	0	Simple ecological system	[291]	S5
SIMP_POP	1	15	1	0	0	Simple population model	[268]	S5
SKIN_D	5	25	30	0	0	Transdermal diffusion (discretized PDE)	[40]	E
SLIP	12	51	4	6	0	Oscillating system with transient influences and slip		E
SLUDGE	12	194	7	0	0	Thermal decomposition of wastewater sludge by thermogravimetry		E
SMALL_EX	2	22	2	0	0	Small example with analytical solution		X
SOOT	2	40	2	0	0	Sooting methane and ethylene flames for LII and Raman measurements	[289]	S5
SPBEDRTD	2	90	9	0	0	Spouted bed reactor mixing model	[147]	S5
SPRING	2	101	2	0	0	Linear spring (unforced harmonic oscillator)	[158]	S0
SS_TUBE	2	6	2	1	1	Diffusion-convection in a tube, steady-state		E
SSHEATEX	2	75	3	0	0	Steady-state, two-pass heat exchanger	[147]	S05
STABCSTR	2	96	2	0	0	Stabilizing nonlinear continuous stirred tank		S5
STABIL	1	21	2	0	0	Stability of chemical reactors with disturbances	[147]	S1
STAGED	4	30	6	0	0	Two-stage culture with product inhibition	[80]	S5
STAR	3	10	4	0	0	Motion of a star within the potential of a cylindrical galaxy		E
STARS	2	183	60	0	0	Gravitational 10-body problem	[178]	S0
STAT5	3	26	4	0	0	Nucleocytoplasmic cycling as a remote censor by phosphorylation of STAT5	[321]	S5
STIFF	5	81	9	0	0	Stiff differential equation	[214]	S5
STIFF_DE	3	30	3	0	0	Stiff ODE	[302]	S5
STIFF_EQ	5	24	2	0	0	Stiff ODE	[361]	S5
STIFF1	3	27	3	0	0	Stiff test problem	[316]	X
STIFF2	4	36	2	0	0	Stiff test problem	[316]	X
SUB_ACCU	4	21	3	0	0	Substrate accumulation with two-phase aerosols		E
SUBTILIS	2	90	7	0	0	Growth and product formation of B. subtilis in batch or continuous culture on molasses	[80]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
SULFUR	9	31	5	1	0	Radioactive sulfur		E
SUPEROX1	3	164	1	0	0	Dismutation of superoxid ion to H2O and O2		E
SUPEROX2	5	198	1	0	0	Catalized reaction of superoxid ion		E
TAB_DIS1	7	94	1	0	0	Immediate release of solid dosage forms, extended model	[195]	E
TAB_DIS2	4	93	1	0	0	Immediate release of solid dosage forms, discrete distribution	[290]	E
TAB_DIS3	5	98	1	0	0	Immediate release of solid dosage forms, lognormal distribution	[195]	E
TAB_DIS4	5	98	1	0	0	Immediate release of solid dosage forms, normal distribution	[195], [290]	E
TAB_DIS5	4	98	1	0	0	Immediate release of solid dosage forms, chi-square distribution	[195], [290]	E
TAB_DIS6	4	98	1	0	0	Immediate release of solid dosage forms, Weibull distribution	[195], [290]	E
TAB_DIS7	5	98	1	1	0	Immediate release of solid dosage forms, Bateman distribution	[195], [290]	E
TAB_DIS8	23	245	5	0	0	Immediate release of solid dosage forms, all distributions	[195], [290]	E
TANK	2	18	1	0	0	Single tank with n-th order reaction	[147]	S1
TANKBLD	2	9	2	0	0	Liquid stream blending	[147]	S5
TANKDIS	1	11	1	0	0	Ladle discharge problem	[147]	S5
TANKHYD	3	40	3	0	0	Interacting tank reservoirs	[147]	S1
TEFLON1	7	129	2	0	0	Viscous-elastic material	[160]	E
TEFLON2	3	451	2	0	0	Viscous-elastic material	[160]	E
TEMPCONT	2	29	2	0	0	Feedback control of a water heater	[80]	S5
TEST_0DE	2	10	1	0	0	Nonlinear ODE with rhs ((a*log(y)+b)/(t+2)-5)*y		S5
THERM	2	36	2	0	0	Thermal stability of a CSTR	[147]	S5
TOLUENE	7	39	3	0	0	Toluene hydrogenation by catalytic reaction	[25], [204]	E
TOPTCSTR	4	240	4	0	0	Time suboptimal control of two-stage continuous stirred tank		S5
TRANSUS1	5	72	6	0	0	Indinavir in Caco-2 without Liposomen and with Vinblastin		E
TRANSUS2	5	66	6	0	0	Indinavir in Caco-2 without Liposomen		E
TREES	2	16	1	0	0	Growth of trees		E

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
TRICHO	4	44	2	0	0	Unstructured growth of trichosporon cutaneum	[16]	S5
TRILEPT	6	157	4	0	0	Estimation of kinetic parameters of a chemical reactor		E
TUBDYN	3	200	8	0	0	Dynamic tubular reactor, eight tanks in series with nth-order reactions	[147]	S5
TUBED	2	19	1	0	0	Axial concentration profile in a tubular reactor	[147]	S5
TUBEMIX	3	54	6	0	0	Non-ideal tube-tank mixing model	[147]	S5
TURBCON	3	90	4	0	0	Turbidostat response	[80]	S5
TWOEX	3	128	4	0	0	Complex two-solute batch extraction with interchanging equilibria	[147]	S5
TWOONE	2	20	3	0	0	Competition between organisms	[80]	S5
TWOSTAGE	2	40	4	0	0	Two-stage chemostat with additional stream	[80]	S5
TWOTANK	3	80	2	0	0	Two tank level control	[147]	S1
UPTAKE	9	51	5	0	0	Compartment model for uptake, metabolic conversion and excretion of a drug		E
US_CSTR	3	72	2	0	0	Unsteady state of a complex reaction in a CSTR	[353]	S5
VAR_META	6	504	8	0	0	Nonlinear biochemical dynamic model with varying metabolite concentration		E
VARMOL	3	78	2	0	0	Gas phase tubular reactor	[147]	S5
VARVOL	4	154	3	0	0	Variable volume fermentation	[80]	S5
VDPOL_0	2	16	2	0	0	Van der Pol equation, electrical circuit	[128]	S0
VDPOL_M	2	18	2	0	0	Van der Pol equation, electrical circuit	[128]	S5
VERT_CHL	8	30	4	6	6	Injected fluid flow in a long, vertical channel (4th order BVP)	[8]	S5
VESSEL	2	5	1	0	0	Vessel adsorption		E
W_WATER1	4	15	2	0	0	Anaerobic wastewater treatment, monod model without decay	[227]	S5
W_WATER2	4	50	2	0	0	Anaerobic wastewater treatment, monod model with substrate inhibition	[227]	S5
WASTEWAT	3	14	5	0	0	Anaerobic wastewater treatment	[236]	S5

(continued)

name	n	$\tilde{l}$	m	m_r	m_e	*background*	*ref*	*data*
WAVE_D1	1	100	10	0	0	Hyperbolic wave equation with exact solution, PDE discretized (Dirichlet)	[277]	S5
WAVE_D2	1	100	10	0	0	Hyperbolic wave equation with exact solution, PDE discretized (Neumann)	[277]	X
WEEDS	8	16	2	0	0	Growth of two weeds in competition	[266]	E
WEIBEL	3	13	1	0	0	Ill-conditioned academic equation		S5
WNDSHEAR	10	7	6	35	0	Optimal take-off trajectories under wind shear	[218]	E
YEASTOSC	1	15	5	0	0	Oscillating continuous Baker's yeast culture	[80]	S5
ZOOPLANK	2	303	3	0	0	Growth of ohytoplankton and zooplankton		S5

3.6 Differential Algebraic Equations

As before, we have r data sets (t_i, c_j, y_{ij}^k) with $l = l_t l_c r$, and l weights w_{ij}^k. Again, weights can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for state variables for which experimental data do not exist. The subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation.

The data fitting function $h(p, y(p,t,c), z(p,t,c), t, c)$ depends on a concentration parameter c and in addition on the solution $y(p,t,c)$ and $z(p,t,c)$ of a system of m_d differential and m_a algebraic equations

$$\begin{aligned}
\dot{y}_1 = \quad & F_1(p,y,z,t,c) && , \; y_1(0) = y_1^0(p,c) \;\; , \\
& \cdots \\
\dot{y}_{m_d} = \quad & F_{m_d}(p,y,z,t,c) && , \; y_{m_d}(0) = y_{m_d}^0(p,c) \;\; , \\
0 = \quad & G_1(p,y,z,t,c) && , \; z_1(0) = z_1^0(p,c) \;\; , \\
& \cdots \\
0 = \quad & G_{m_a}(p,y,z,t,c) && , \; z_{m_a}(0) = z_{m_a}^0(p,c) \;\; .
\end{aligned}$$

Without loss of generality, we assume that the initial time is zero. Now $y(x,t,c)$ and $z(x,t,c)$ are solution vectors of a joint system of $m_d + m_a$ differential and algebraic equations (DAE). The initial values of the differential equation system $y_1^0(p,c)$, ..., $y_{m_d}^0(p,c)$ and $z_1^0(p,c)$, ..., $z_{m_a}^0(p,c)$ may depend on one or more of the system parameters to be estimated, and on the concentration parameter c.

The system of differential equations is called an index-1-problem or an index-1-DAE, if the algebraic equations can be solved with respect to z, i.e., if the matrix

$$\nabla_z G(p,y,z,t,c)$$

possesses full rank. In this case, consistent initial values are can be computed internally.

The resulting parameter estimation problems is

$$p \in I\!R^n : \quad \begin{aligned}
\min \quad & \textstyle\sum_{k=1}^{r} \sum_{i=1}^{l_t} \sum_{j=1}^{l_c} (w_{ij}^k (h_k(p, y(p,t_i,c_j), z(p,t_i,c_j), t_i, c_j) \;\; - \;\; y_{ij}^k))^2 \\
& g_j(p) = 0 \;\; , \;\; j = 1, ..., m_e \;\; , \\
& g_j(p) \geq 0 \;\; , \;\; j = m_e + 1, ..., m_r \;\; , \\
& p_l \leq p \leq p_u \;\; .
\end{aligned}$$

We assume that the model functions $h_k(p,y,z,t,c)$ and $g_j(p)$ are continuously differentiable functions of p, $k = 1$, ..., r and $j = 1$, ..., m_r, and that the state variables $y(p,t_i,c_j)$ and $z(p,t_i,c_j)$ are smooth solutions subject to p. All test problems based on differential algebraic equations are listed in Table 5, where constraint counts are omitted.

Table 5. Differential Algebraic Equations

name	n	$\tilde{l}$	m_d	m_a	*background*	*ref*	*data*
2LKC_ROB	5	80	12	5	Two-link planar robot with constraints	[8]	S5
AEROSOL	4	29	2	2	Substrate concentration in two-phase aerosol devices		E
AEROSOLS	5	36	2	2	Substrate concentration in two-phase aerosol devices, simulated data		S1
BATCH	9	128	6	3	Isothermal batch reactor, slow and fast reactions	[31]	S5
BATCH_E	9	204	6	4	Isothermal batch reactor, slow and fast reactions, two data sets	[31]	E
BATCH_E1	9	131	6	4	Isothermal batch reactor, slow and fast reactions, data for 40 deg C	[31], [319]	E
BATCH_E2	9	84	6	1	Isothermal batch reactor, slow and fast reactions, data for 67 deg C	[31], [319]	E
BATCH_E3	9	124	6	1	Isothermal batch reactor, slow and fast reactions, data for 100 deg C	[31], [319]	E
BATCH_F1	9	120	6	4	Isothermal batch reactor, slow and fast reactions, one data set (313,15)	[31]	E
BATCH_F2	9	84	6	4	Isothermal batch reactor, slow and fast reactions, one data set (340,15)	[31]	E
BATCH_F3	9	204	6	4	Isothermal batch reactor, slow and fast reactions, two data sets	[31]	E
BATCH_F4	9	328	6	4	Isothermal batch reactor, slow and fast reactions, three data sets	[31]	E
BATCH_X1	17	120	6	4	Isothermal batch reactor, slow and fast reactions, experimental design	[31]	E
BATCH_X2	13	78	6	4	Isothermal batch reactor, slow and fast reactions, experimental design	[31]	ED
BATCHREA	5	66	6	1	Batch reactor	[53]	S5
BOND	4	24	2	1	Transition of photon in a hydrogen-hydrogen bond	[176]	S5
BUBBLEC	3	72	8	5	Bubble point calculation for a batch distillation column	[147]	S5
CAT_SD	6	300	4	4	Catalyst semiconductor		E
CELLS	5	120	3	2	Cultivation of isolated plant cells in suspension culture	[226]	S1
CONDENS	2	114	1	5	Condensation of methanol with constant volume	[245]	S5
COPPER1	5	277	6	1	Multilayer model for moisture adsorbed on copper		E

(continued)

name	n	$\tilde{l}$	m_d	m_a	*background*	*ref*	*data*
DAE_EX	2	12	3	0	DAE with singularity		S5
DAE_I1	2	100	4	1	Academic example, index-1-formulation		S0
DAE_I2	2	40	4	1	Academic example, index-2-formulation		S0
DAE_I3	2	40	4	1	Academic example, index-3-formulation		S0
DAE_IN2	2	26	2	1	System of three differential algebraic equations with index 2	[8]	S5
DAE_IN2X	4	30	2	1	System of 3 differential algebraic equations with index 2, exact solution	[8]	X
DAE_SYS	2	20	1	2	Particle diffusion and reaction (2nd order BVP)	[8]	S1
DISTILL	3	66	22	22	Distillation column for two substances	[124]	S1
DISTILL3	9	205	106	53	Distillation column for three substances		S1
EVAPOR	3	38	3	10	Evaporation of benzol with constant volume	[208]	S5
EXOBATCH	4	80	5	6	Batch reactor with strongly exothermic reactions and cooling jacket	[345]	S5
HYDRODYN	2	15	2	1	Unipolar hydrodynamic model for semiconductors in the isotropic case	[8]	S5
LKIN_BR	2	34	2	0	Simple linear compartment model with two break points		S5
MARBLE	2	202	6	1	Movement of marble in a bowl		S5
MEM_WIRE	5	40	3	1	Optimal form of shape memory wires		S1
P_IDENT	6	138	2	1	Identification of parameters, academic example	[207]	S5
P_IDENT1	4	138	2	1	Identification of parameters, academic example, reduced parameter set	[207]	S5
P_IDENT2	4	19	2	1	Identification of parameters, academic example, one data set and u1=u2=1	[207]	S5
P_IDENT3	8	19	2	1	Identification of parameters, academic example, one data set and experimental design	[207]	ED
PENDU_CS	2	80	8	2	Plain pendulum, index-3-formulation with two algebraic equations		S1
PENDU_I1	2	80	4	1	Plain pendulum, index-1-formulation		S1
PENDU_I2	2	80	4	1	Plain pendulum, index-2-formulation		S1
PENDU_I3	2	80	4	1	Plain pendulum, index-3-formulation		S1
PENDU_IV	2	80	4	3	Plain pendulum, consistent initial values computed internally		S1
PENDU_PD	2	80	4	3	Plain pendulum, projected descriptor formulation		S1

(continued)

name	n	$\tilde{l}$	m_d	m_a	*background*	*ref*	*data*
PENDULUM	2	80	4	1	Plain pendulum		S1
PHOSPH_A	3	54	3	2	Chemical reaction, phosphorescence		S5
RESPIR	3	40	1	1	Human respiratory system	[348]	S5
SCARA	4	204	8	2	Scara robot with four parameters and given control function	[136]	N5
SCARA_X	24	66	8	2	Scara robot with experimental design	[136]	ED
SHOCK	4	60	6	3	Reaction zone in detonating explosives	[84]	S5
TRAN_A_B	4	102	2	1	Transport of substrate by Michaelis-Menten kinetics		S1
TRANSIST	3	11	3	2	Transistor amplifier, highly oscillating		E
TRUCK	3	84	22	1	Truck model (multibody system)	[308]	S1
TUBULAR	8	42	2	2	Stationary tubular reactor with cooling wall	[214]	S5
URETHAN	15	30	3	3	Urethan reaction in a semi batch reactor with two feed vessels	[22]	E
URETHANW	55	164	3	3	Urethan reaction in a semi batch reactor, experimental design with weights	[22]	ED
URETHANX	55	40	3	3	Urethan reaction in a semi batch reactor, experimental design	[22]	ED
VAS_ADSP	4	14	2	1	Vascular adsorption		S5
VAS_ADSS	13	34	6	3	Vascular adsorption with three experimental data sets		E
VDPOL	2	10	1	1	Van der Pol equation, electrical circuit		S0

3.7 Partial Differential Equations

Now we proceed from r data sets

$$(t_i, y_i^k)\ ,\ i = 1, \ldots, l_t,\ k = 1, \ldots, r\ \ ,$$

where l_t time values and $l = l_t r$ corresponding measurement values are defined. Moreover, we assume that l weights w_i^k are given, which can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for state variables for which experimental data do not exist. The subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation. The additional independent model variable c called concentration in the previous models, is not taken into account for simplicity.

The system of partial differential equations under consideration is

$$\begin{aligned} \dot{u}_1 &= F_1(p, u, u_x, u_{xx}, v, x, t)\ \ , \\ &\ldots \\ \dot{u}_{n_p} &= F_{n_p}(p, u, u_x, u_{xx}, v, x, t) \end{aligned}$$

with state variable $u = (u_1,\ \ldots,\ u_{n_p})^T$. We denote the solution of the system of partial differential equations by $u(p, x, t)$ and $v(p, t)$, since it depends on the time value t, the space value x, and the actual parameter value p. v denotes the additional coupled differential variable. To simplify the notation, flux functions are omitted.

Initial and boundary conditions may depend on the parameter vector to be estimated. Since the starting time is assumed to be zero initial values, have the form

$$u(p, x, 0) = u_0(p, x)$$

and are defined for all $x \in (x_L, x_R)$. For both end points x_L and x_R we allow Dirichlet or Neumann boundary conditions

$$\begin{aligned} u(p, x_L, t) &= u^L(p, v, t)\ \ , \\ u(p, x_R, t) &= u^R(p, v, t)\ \ , \\ u_x(p, x_L, t) &= \hat{u}^L(p, v, t)\ \ , \\ u_x(p, x_R, t) &= \hat{u}^R(p, v, t) \end{aligned}$$

for $0 < t \leq T$, where T is the final integration time, for example the last experimental time value t_{l_t}. We do not require the evaluation of all boundary functions. Instead, a user may omit some of them depending upon the structure of the PDE model, for example whether second partial derivatives exist in the right-hand side or not.

In addition, the partial differential equation may depend on the solution of a system of ordinary differential equations $v \in I\!R^{n_c}$ given in the form

$$\dot{v}_j = G_j(p, u(p, x_j, t), u_x(p, x_j, t), u_{xx}(p, x_j, t), v, t)$$

for $j = 1, \ldots, n_c$, where $u(p, x, t)$ is the solution vector of the partial differential equation. Here x_j are any x-coordinate values where the corresponding ordinary differential equation is coupled to the partial one. Some of these values may coincide. When discretizing the system by the method of lines, they are rounded to the nearest neighboring grid point. The corresponding initial values

$$v(p, 0) = v_0(p)$$

may depend on the parameters to be estimated.

Each set of experimental data is assigned a spatial variable value $x_k \in (x_L, x_R)$, $k = 1,\ \ldots,\ r$, where r denotes the total number of measurement sets. Some or all of the x_k-values may coincide, if different measurement sets are available at the same local position. Since partial differential equations

are discretized by the method of lines, the fitting points x_k are rounded to the nearest spatial grid point or line, respectively.

The resulting parameter estimation problems is

$$\begin{aligned} \min \quad & \sum_{k=1}^{r} \sum_{i=1}^{l_t} (w_i^k (h_k(p, u(p, x_k, t_i), u_x(p, x_k, t_i), \\ & \qquad u_{xx}(p, x_k, t_i), v(p, t_i), t_i) \;-\; y_i^k))^2 \\ p \in I\!R^n : \quad & g_j(p) = 0 \;, \quad j = 1, ..., m_e \;\;, \\ & g_j(p) \geq 0 \;, \quad j = m_e + 1, ..., m_r \;\;, \\ & p_l \leq p \leq p_u \;\;. \end{aligned}$$

It must be assumed that all model functions $h_k(p, u, u_x, u_{xx}, v, t)$ and $g_j(p)$ are continuously differentiable subject to p for $k = 1, \ldots, r$ and $j = 1, \ldots, m_r$, also the state variables and their spatial derivatives $u(p, x, t)$, $u_x(p, x, t)$, $u_{xx}(p, x, t)$, and $v(p, t)$.

All test problems of our collection based on time-dependent, one-dimensional partial differential equations are listed in Table 6. Not listed are the number of integration areas, switching times, and structure of the boundary conditions. Equality constraints do not exist in this case, and m_e is therefore omitted.

Table 6. Partial Differential Equations

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
2AREAS	5	24	1	0	0	Diffusion (Fick's law) in two areas	[278]	S5
2AREAS_C	4	101	1	1	0	Diffusion (Fick's law) in two areas and coupled ODE		S5
2MEMBRAN	4	26	1	3	0	Two membranes with explicitly modeled transition		E
ACCRET	1	40	2	0	0	Thermal equilibrium curves in Keplerian accretion disks	[252]	S5
ADV_DIFF	2	150	3	0	0	Advection-diffusion equation with Riemann initial data	[156]	S1
ADV_DOM	2	150	1	0	0	Advection dominated equation, wave from left boundary	[314]	S5
ADV_DOMS	2	48	3	0	0	Advection dominated equation, wave from left boundary, with sensitivity equations	[314]	S1
ADV_VC	2	60	1	0	0	Advection equation with variable coefficient	[320]	X
ADV_VTC	3	90	2	0	0	Advection equations with variable time coefficient	[320]	S5
ADVEC_2N	3	50	1	0	0	Nonlinear unsteady advection (n=2)	[352]	S1

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
ADVEC_5N	3	20	1	0	0	Nonlinear unsteady advection (n=5)	[352]	S5
ADVEC_CP	2	20	2	2	0	Two coupled advection equations with periodic boundary conditions	[169], [352]	X
ADVEC_LU	4	35	1	0	0	Linear unsteady advection-diffusion	[352]	S5
ADVEC_PB	1	50	1	1	0	Advection with periodic boundary condition	[169], [352]	S5
ADVEC_S	4	5	1	0	0	Linear steady advection-diffusion with source term	[352]	S5
ADVECT	1	171	1	0	0	Advection equation, first-order hyperbolic PDE	[277]	X
ADVECT_N	1	128	1	0	0	Advection equation with a nonlinear source term	[248]	S5
ADVECT_R	1	180	1	0	0	Advection equation with right boundary value	[277]	X
ADVECT_S	1	128	2	0	0	Advection with a nonlinear source term and sensitivity equations	[248]	S5
ADVECT2	2	190	2	0	0	Two advection equations (different flux directions)	[277]	X
ADVECT2A	2	190	2	0	0	Two advection equations (different flux directions, two areas)	[277]	X
AFFIN	4	60	2	0	0	Affinity membrane separation of a protein solution	[54]	S5
AIR_FLOW	1	32	3	0	0	Flow of air in shock-tube (Euler equations of gas dynamics, Riemann data)	[248]	S1
AL_ALLOY	2	20	1	0	0	Finite heat-conducting in aluminium alloy	[24], [94]	E
ALDRIN_P	4	10	1	0	0	Diffusion with chemical reaction		E
ARA_YARN	3	120	1	0	0	Water penetration into an aramide yarn	[319]	S5
AX_DIFF	2	11	1	0	0	Continuous tubular reactor with axial diffusion	[57]	S0
BALLONS5	5	26	1	5	0	Diffusion of air through skins of 5 bowls		E
BEETLES	2	80	1	0	0	Flea beetles in cultivated linear arrays of collard patches (insect dispersal study)	[19]	S0
BINDSITE	6	50	6	0	0	Plasma, extravascular, and binding sites with two injections		E

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
BIOFILM	3	20	2	2	0	Double substrate biofilm reaction	[80]	S5
BLD_BRN	10	14	1	6	0	Blood-brain barrier		E
BLOW_UP	1	49	1	0	0	Degenerated parabolic equation with blow-up	[105]	S5
BRAIN	3	90	1	0	0	Transport phenomena in brain tissue	[19]	S5
BRINE	1	15	2	0	0	Brine transport in porous media	[367]	X
BRUSS2D	2	220	42	0	0	Two-dimensional brusselator	[128]	S5
BRUSSEL	3	120	2	0	0	Brusselator with diffusion	[102], [284]	S5
BSE	4	10	1	0	0	Black-Scholes equation governing price of derivative security	[36], [363]	S1
BUBB_BIO	1	10	3	0	0	Bubble column bio-reactor	[225]	S5
BUBBLE	2	108	2	0	0	Dynamic oxygen uptake of water in bubble column		S5
BURGER	1	24	1	0	0	Parabolic Burger's equation with exact solution	[309]	X
BURGER_E	1	120	1	0	0	Viscous Burger's equation with exact solution, mue=0.01	[277], [284]	X
BURGER_F	1	20	1	0	0	Viscous Burger's equation with exact solution, mue=1	[277], [284]	X
BURGER_I	3	32	1	0	0	Burger's equation in the inviscid limit	[365], [248]	S5
BURGER_R	1	44	1	0	0	Burger's equation with Riemann initial data	[309]	X
BURGER_X	3	50	1	0	0	Viscous Burger's equation with exact solution, eps=0.0005		X
BURST	1	105	2	0	0	Crisis induced intermittent bursting in reaction-diffusion chemical systems	[91], [89]	S5
CALIBR	12	195	4	4	0	Substrate diffusion through two areas		E
CARRIER	3	12	3	2	0	Diffusion through membrane based on carrier effect		S5
CD_TRANS	1	28	1	0	0	Convective-dispersive transport equation with nonlinear reactions	[163]	S5
CN_PLAS	2	30	5	0	0	Magnetohydrodynamic model of a charge-neutral plasma	[341]	S5
CNT_CUR1	4	80	2	0	0	Counter-current separation of fluid phase concentrations with phase equilibrium	[250]	S5
COLLAGEN	2	8	2	0	0	Drug release from collagen matrices	[259]	S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
COMP_MED	3	10	1	0	0	Infinite composite medium	[65]	S1
CON_DIV1	1	90	1	0	0	Periodic convection dominated diffusion	[210]	X
CON_DIV2	1	63	1	0	0	Periodic convection dominated diffusion	[210]	X
CONTAMIN	3	24	4	4	0	Contamination of aqueous solutions	[284]	S5
COS_PROF	1	171	1	0	0	Convection equation, propagation of a cosinus profile	[277]	S5
CPA_PLAS	3	12	1	2	0	CPA plaster with three diffusion areas		E
CPL_ADV	2	90	2	2	0	Two coupled linear advection equations	[189]	S1
CRYSTAL	4	51	2	0	0	Crystal dissolution fronts in flows through porous media	[161]	S5
CSE	2	41	2	0	0	Cubic Schroedinger equation with one soliton	[278]	S5
CTFLOW_P	1	44	2	0	0	Two incompressible countercurrent flows of binary liquid mixture with permeable wall	[222]	S5
CTRL_WAV	20	40	2	0	0	Optimal control problem, wave equation		S5
CUBIC	1	10	1	0	0	Cubic conservation law with Riemann data	[132]	S5
DAMBREAK	2	30	2	0	0	Idealized dam break, sudden and complete removal	[314]	S5
DBVP	2	9	1	0	0	Dirichlet boundary value problem with dominating heat source	[61]	S5
DC_TUBE	1	7	1	0	0	Diffusion-convection in a tube		S1
DEHYDRO	4	60	2	0	0	Dehydrogenization of ethylbenzene to styrene in a tubular reactor	[353]	S05
DERMAL	10	25	2	4	0	Transdermal skin model in two areas with transitions	[315]	E
DESIGN	1	56	1	0	0	First-order hyperbolic PDE, inhomogeneous part	[334]	S1
DIALYSI1	4	179	1	1	0	Dialysis membrane with exponential diffusion coefficient, long term experiment		E
DIALYSI2	2	96	1	2	0	Substrate diffusion through dialysis membrane		E
DIALYSI3	3	100	1	1	0	Substrate diffusion through dialysis membrane with two areas		E

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
DIALYSI4	8	298	3	3	0	Substrate diffusion through dialysis membrane with 2 areas, 3 data sets		E
DIALYSI5	7	298	3	3	0	Substrate diffusion through dialysis membrane with 2 areas, 3 data sets		E
DIFF_1D	4	151	1	0	0	Diffusion problem with Dirichlet and Neumann boundary conditions		E
DIFF_ADS	3	50	2	0	0	Diffusion and absorption reaction		E
DIFF_CON	2	40	1	0	0	Diffusion-convection problem with discontinuous coefficients	[309]	S5
DIFF_ETH	2	50	1	0	0	Diffusion of ethanol in water	[353], [137]	S5
DIFF_NLB	4	35	1	0	0	Nonlinear diffusion with nonlinear boundary condition	[309]	S5
DIFF_P	3	12	1	2	0	Flow with diffusion through tube wall		S5
DIFFPT	3	10	1	0	0	Diffusion and partitioning in biological systems, non-continuous transition	[219]	S1
DIFFPT1	3	20	1	0	0	Diffusion and partitioning in biological systems, exponential initial values	[219]	S5
DIFFPT2	3	10	1	0	0	Diffusion and partitioning in biological systems, non-continuous transition	[219]	S1
DIFFPT3	4	20	1	0	0	Diffusion and partitioning in biological systems, continuous flux transition	[219]	S5
DIFFPT4	4	20	1	0	0	Diffusion and partitioning in biological systems, continuous flux	[219]	S5
DIFFREA	3	292	1	1	0	Diffusion and reaction in solid phase, coupled ODE		E
DIFFREA1	9	292	1	0	0	Diffusion and reaction in solid phase		E
DIFFUS	1	81	1	0	0	Diffusion equation with constant parameters		S5
DIGESTOR	2	31	1	0	0	Fixed-bed anaerobic digestor		S5
DISADVFL	2	22	1	0	0	Dispersive advective flow		E
DISRE	3	30	1	0	0	Non-isothermal tubular reactor with axial dispersion	[147], [284]	S5
DISRET	2	12	2	0	0	Non-isothermal tubular reactor with axial dispersion and non-constant reaction term	[147], [284]	S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
DRY	2	46	1	2	0	Drying of a solid	[147]	S1
ECOLOGY	2	60	2	0	0	Population ecology with planktonit predator-prey and crowding	[173]	S5
ELASTIC	1	20	2	0	0	Elastic model in conservative form with discontinouity	[64]	S5
ELEC_DYN	3	20	2	0	0	Electrodynamical application	[37]	S5
ELECTRO	3	38	2	0	0	Electrodynamic model	[284]	S5
ELLIPTIC	3	42	1	0	0	Elliptic test problem	[309]	S5
ENERGY	2	5	1	0	0	Tubular reactor based on energy equation	[115]	S5
ENZDYN	3	22	2	0	0	Dynamic diffusion and enzymatic reaction	[147]	S5
EXR14_1	3	10	1	0	0	Signaling problem	[190]	S5
EXR14_2	1	13	1	0	0	Initial value problem with non-continuous initial value	[190]	S5
EXR14_3	3	14	1	0	0	Initial-boundary value problem	[190]	S5
EXR15_4	1	30	2	0	0	Traveling wave	[190]	S5
EXR21_1	2	100	1	0	0	Initial value problem	[190]	S5
EXR22_1	2	10	1	0	0	Initial value problem	[190]	S5
EXR23_1	2	10	1	0	0	Initial value problem	[190]	S5
EXR24_1	2	13	1	0	0	Nonlinear initial value problem	[190]	S5
EXR25_1	2	20	1	0	0	Initial value problem	[190]	S5
EXR25_2	2	10	1	0	0	Nonlinear initial value problem	[190]	S5
EXR25_6	2	8	1	0	0	Nonlinear initial value problem	[190]	S5
EXR31_4	2	11	1	0	0	Initial value problem with shock formation	[190]	X
EXR32_1	2	11	1	0	0	Initial value problem, breaking wave	[190]	S5
EXR32_4	2	10	1	0	0	Initial-boundary value problem	[190]	S5
EXR32_7	2	28	1	0	0	Initial value problem, formation of wave	[190]	S5
EXR32_8	2	8	1	0	0	Initial value problem with shock path	[190]	S5
EXR32_9	2	24	1	0	0	Initial value problem with two shocks merging into one shock	[190]	S5
EXR34_12	2	10	1	0	0	Riemann problem	[190]	S5
EXR42_2	2	10	1	0	0	Initial-boundary value diffusion problem	[190]	S5
EXR43_2	2	10	1	0	0	Nonlinear diffusion problem	[190]	S5
EXR43_3	2	10	1	0	0	Diffusion problem	[190]	S5
EXR52_5	2	28	2	0	0	Hyperbolic initial value problem, linear	[190]	S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
EXR64_4	2	10	1	0	0	Diffusion problem	[190]	S5
EXR65_5	1	10	1	0	0	Diffusion problem (no global solution for a¿1)	[190]	S5
FILTWASH	2	20	1	0	0	Filter washing	[147]	S5
FINAG	2	246	2	0	0	Nerve conduction	[229]	S5
FIX_BED	2	16	2	0	0	Catalytic fixed bed reactor	[258]	S5
FIXBED	3	48	2	0	0	Catalytic fixed bed reactor with one exothermal reaction	[336], [87]	S1
FLAME	2	48	2	0	0	Dwyer-Sanders flame propagation model	[81], [346], [284]	S1
FLOW	2	18	1	0	0	Isothermal laminar-flow tubular reactor	[335]	S5
FLOW_PMD	1	80	1	0	0	Flow through porous media with degenerate initial values	[337]	S1
FLOW_PMW	3	120	1	0	0	Flow through porous media with waiting time	[337]	S5
FLUID	3	20	2	0	0	Diffusion (Fick's law)	[278]	S5
FOX	2	189	3	0	0	Rabies distribution of fox population	[228]	S1
FRONT	2	60	2	0	0	Flame propagation model with non-constant moving front	[249]	S5
G_HILL	2	33	1	0	0	Convection-diffusion of a Gaussian hill	[79]	S5
GAS_BUBB	2	180	1	14	0	Non-viscous gas bubble in oil with diffusion		S5
GAS_CONV	3	243	1	0	0	Gas convection		E
GAS_DIF1	1	8	1	2	0	One-dimensional gas diffusion in a column		E
GAS_DIF2	1	15	1	2	0	One-dimensional gas diffusion in a column		E
GLACIER	2	45	1	0	0	Glacier growth with conservation of mass and momentum, incompressible flow	[156]	S5
GLYCO	2	62	2	0	0	Glycolysis reaction-diffusion model with autocatalytical growth of species	[9]	S1
GROUND_W	5	20	1	0	0	Saturation of ground water (Richards equation)	[338], [284]	E
GROWTH	2	21	1	0	0	Logistic model of population growth (Fisher's equation)	[332]	S5
HEAT	1	6	1	0	0	Heat equation	[277]	S5
HEAT_B	6	36	1	0	0	Heat equation, break points and two integration areas with transition condition		S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
HEAT_BD3	2	32	1	0	0	Nonlinear heat equation, boundary conditions of third type		S5
HEAT_CD	2	10	1	0	0	One-dimensional heat conduction	[280]	S5
HEAT_CF	5	11	1	0	0	Heat transfer in a circular fin	[280], [33]	S5
HEAT_CON	3	76	1	0	0	Heat transfer in cylinder with heat loss by convection	[353]	S5
HEAT_CW	2	10	1	0	0	Graetz problem with constant wall heat flux	[280]	S5
HEAT_CYL	2	50	1	0	0	Cylindrical heat transfer	[309]	S5
HEAT_EX	3	20	1	0	0	Tubular heat exchanger	[280]	S5
HEAT_I	4	40	1	0	0	Heat equation, two integration areas with transition condition		S5
HEAT_MS	2	9	2	0	0	Heat transport equation at the microscale (3rd order)	[358]	X
HEAT_NLB	2	10	1	0	0	Heat equation with nonlinear boundary condition of Stefan-Boltzmann type	[331]	E
HEAT_S1	4	21	1	0	0	Heat equation with redundant parameters	[277], [284]	S5
HEAT_S2	3	11	1	0	0	Heat equation with redundant parameters	[277], [284]	S5
HEAT_SEN	3	40	4	0	0	Heat conduction with full sensitivity equations		S5
HEAT_SX	1	20	2	0	0	Heat equation with one sensitivity equation and exact solution		X
HEAT_TDC	3	42	1	0	0	Heat diffusion with time-dependent diffusion parameter		S5
HEAT_X	1	99	1	0	0	Heat equation with exact data and maximum norm		X
HOT_SPOT	2	110	1	0	0	'Hot Spot' problem from combustion theory	[346], [284]	S5
HUMID	2	33	3	0	0	Humidification column of porous medium	[277]	S1
HYDRO	1	32	2	0	0	St. Venant equation for fluid dynamics of hydro systems	[122]	S5
HYDRO_2C	6	20	2	0	0	St. Venant equation for fluid dynamics of hydro systems, two serial channels	[122]	X
HYDRO_3S	6	303	6	0	0	St. Venant equation for fluid dynamics of hydro systems, 3-star		none
HYDRO_FX	1	32	2	0	0	St. Venant equation for fluid dynamics of hydro systems, flux formulation	[122]	S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
HYG_POLY	3	74	1	0	0	Diffusion of water into a hygroscopic polymer		E
HYGROS	3	9	1	0	0	Diffusion of water through boundary layer of hygroscopic material and air		E
HYP_PBC	2	20	1	1	0	Hyperbolic equation with periodic boundary conditions	[9]	S5
HYP2ND	1	15	2	0	0	Hyperbolic equation of second order, alternating cosine waves		S5
HYPER	2	198	2	0	0	System of two advection equations, first-order hyperbolic PDEs		S5
HYPERBO1	2	90	2	0	0	Hyperbolic test system	[18]	S5
HYPERBO2	2	90	2	0	0	Hyperbolic test system	[18]	S5
HYPERBO3	2	90	2	0	0	Hyperbolic test system	[18]	S5
HYPERBO4	2	90	2	0	0	Hyperbolic test system	[18]	S5
HYPERBO5	3	90	2	0	0	Hyperbolic test system	[18]	S5
IN_LAYER	4	42	2	0	0	Catalyst with inert layers (diffusion, absorption, desorption)		S5
INTEG	3	25	1	0	0	Population dynamics with integro-differential equation	[248], [284]	E
INTERF1	2	10	1	0	0	System with interface (not modeled)	[284]	S5
INTERF2	2	18	1	0	0	System with interface	[284]	S5
INV_PROB	10	20	1	0	9	Inverse problem in heat conduction	[131]	S5
IONTRAN1	2	101	1	2	0	Ion transport through membrane by diffusion		S1
IONTRAN3	3	96	1	1	0	Ion transport through membrane with Langmuir isotherm for sorption		E
ISOTHRM1	10	34	2	0	9	Reactive solute transport, advective-dispersive transport	[146], [145]	S5
ISOTHRM2	4	20	1	0	3	Reactive solute transport, advective-dispersive transport (Freundlich version)	[146], [145]	S5
JONTO	4	17	1	1	2	Optimal control of iontophoresis with three membranes		none
KILN	10	22	1	0	0	Heating a probe in a kiln	[72]	E
KIN_SORP	2	101	2	0	0	Kinetic sorption by advection-dispersion		S5
LAM_FLOW	1	10	1	0	0	Unsteady laminar flow in a circular tube	[280], [143]	S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
LAPLACE	1	20	20	0	0	Laplace equation (elliptic)	[277]	X
LDCP	1	100	1	0	0	Linear diffusion-convection equation	[257], [250]	S5
LIN_ADV	4	90	1	0	0	Linear advection problem, highly nonlinear initial condition	[150]	S5
LIN_HC	3	140	1	0	0	Linear heat conduction	[2]	X
LIN_HYP1	2	25	1	0	0	First-order linear hyperbolic equation	[355]	X
LIN_HYP2	3	70	1	0	0	First-order linear hyperbolic equation with interface	[355]	X
LIN_HYP3	2	212	1	0	0	First-order linear hyperbolic equation with variable velocity field	[355]	X
LNCHROM1	3	1	1	0	0	Nonlinear chromatographic system	[145]	S5
LNCHROM2	2	282	4	0	0	Nonlinear chromatographic system		S5
LOSSLESS	2	20	2	0	0	Lossless electric transmission line	[278]	X
LUNG	5	33	1	4	0	Protein application in lung with decomposition		E
MALTDX10	14	114	10	10	0	Drying of maltodextrin in a convection oven, simultaneous fitting of 10 data sets	[104]	E
MALTODEX	5	12	1	1	0	Drying of maltodextrin in a convection oven, first data set	[104]	E
MASS_TRA	1	25	1	0	0	Mass transfer with simultaneous convection and diffusion	[280]	S5
MEDAKZO	2	21	2	0	0	Medical Akzo-Nobel problem	[187]	S5
MEM_SEP	3	25	2	0	0	Affinity membrane separation of a protein solution	[53]	S5
MEMBRANE	3	20	2	0	0	Diffusion through a membrane	[232]	S5
MILL1	1	80	1	0	0	Rolling mill cooling, constant psi in boundary condition	[284]	S0
MILL2	5	40	1	0	0	Rolling mill cooling, variable phi in boundary condition	[284]	S1
MILL3	2	80	1	0	0	Rolling mill cooling, estimating heat transfer coefficients	[284]	S1
MOL_DIFF	2	10	1	0	0	Molecular diffusion (boundary value problem)	[217]	E
MOVFRONT	3	126	1	0	0	Moving front (Burger's equation)	[2]	S5
MX_ENTRO	9	80	1	0	0	Maximum entropy method, advection-diffusion equation		S5

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
MZ_FURN	5	63	2	0	0	Multizone electrical furnace for production of integrated circuits	[364]	S05
N_CONVEX	1	36	1	0	0	Hyperbolic test problem of Shu and Osher, nonconvex flux	[149]	S5
NDYN	3	20	2	0	0	Nitrogen and ammonium dynamics in forest soils	[51], [284], [101]	E
NERVE	4	200	2	0	0	Nerve pulse	[347]	S5
NL_HEAT	3	18	1	0	0	Nonlinear heat equation	[332]	S5
NL_PDE	3	10	1	0	0	Highly nonlinear PDE with exact solution	[278]	X
NL_PDE1	1	10	1	0	0	First order nonlinear PDE with exact solution		S5
NL_TRANS	2	11	1	1	0	Nonlinear transport equation developing a shock (Burger), periodic boundary	[282]	S5
NL2_SORP	3	38	2	0	0	Nonlinear two-point sorption		E
NLIN_2ND	2	100	1	0	0	Nonlinear second-order partial derivatives		S5
NLINPDE	2	100	2	0	0	Two nonlinear PDE's with exact solution	[277], [206]	X
NLSE	4	18	2	0	0	Nonlinear Schroedinger equation, exact soliton solution (complex)		S1
NOISE	2	20	1	0	0	Nonlinear deblurring and noise removal	[211]	S5
NON_AD	1	60	1	0	0	Nonlinear advection-diffusion equation	[156]	S5
OBSTACLE	2	20	2	0	0	Shallow water flow over an obstacle	[188]	S1
ONESTEP	2	130	2	0	0	One-step reaction with diffusion and non-unit Lewis number	[2]	S1
OSC_SOL	2	20	3	0	0	Oscillatory solution of hyperbolic PDE	[92]	S1
PACK_BED	2	64	4	0	0	Fluid through a packed bed with adsorption/desorption of two components		S5
PAR_CTRL	2	10	1	0	0	Parabolic optimal control problem	[220]	X
PAR_SIN	2	84	1	0	0	Parabolic PDE with inhomogeneous sinus-term	[262], [251]	X
PARAB1	3	8	1	0	0	Parabolic equation, identifiability test	[19]	S0
PARAB2	3	60	2	0	0	Parabolic equation, identifiability test	[19]	S0

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
PARAB3	2	30	1	0	0	Parabolic equation, identifiability test	[19]	S0
PARAB4	3	30	1	0	0	Parabolic equation, identifiability test	[19]	S0
PARAB5	3	10	1	0	0	Parabolic equation, identifiability test	[19]	S0
PARAB6	6	27	1	0	0	Parabolic equation, identifiability test	[19]	X
PB_CTRL1	51	98	1	0	0	Parabolic optimal control problem	[112]	none
PB_CTRL2	11	48	1	0	0	Parabolic optimal control problem	[112]	none
PERIOCHP	2	12	3	0	0	Diffusion-mediated release from bulk degrading matrices in dental processes	[333]	E
PHYP_PBC	2	20	1	1	0	Parabolic-hyperbolic equation with periodic boundary conditions	[9]	S5
POLLUTN	8	28	4	0	0	SST pollution in the stratosphere	[309], [284]	S1
POLY_DYN	3	15	1	1	0	Chain length of polymerization process		S5
POLYMERI	9	45	12	0	0	Radical copolymerization of methylmethacrylat and styren	[296]	S1
POOL	3	28	2	1	0	Evaporation of vapor from a pool of liquid	[27]	S5
PORE	7	51	2	1	0	Diffusion through polymer pores		S5
QUENCH1	2	63	1	0	0	Degenerate nonlinear quenching	[307]	S5
QUENCH2	1	6	1	0	0	Degenerate nonlinear quenching	[307]	S5
REA_DIF1	1	45	1	0	0	Reaction-diffusion equation	[88]	S5
REA_DIF2	2	135	1	0	0	Reaction-diffusion equation	[88]	S5
RESERVOI	1	10	1	0	0	Reservoir simulation by the Buckley-Leverett equation	[341]	S5
RICH_EQU	4	120	1	0	0	Saturation of ground water (Richards equation)	[338], [264]	S1
RICH_PT	5	202	1	1	0	Non-stationary fluid transport through porous media by Richards equation	[35]	S5
RICH_XEN	3	36	1	0	0	Saturation of ground water (Richards equation)	[338], [264]	S1
RIE_BND	4	20	3	0	0	Flow of air in shock-tube (Euler equations of gas dynamics), flux formulation	[248]	S1

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
RIE_CD	1	33	3	0	0	Riemann problem for Euler equations of a polytropic gas with contact discontinuity	[189]	S1
RIE_LAX	1	6	3	0	0	Riemann problem for Euler equations, formulation of Lax	[189]	S5
RIE_LD	1	33	3	0	0	Low density and internal energy Riemann problem for Euler equations	[189]	S1
RIE_SEWI	2	60	3	0	0	Riemann problem with shock entropy wave interaction for Euler equations	[150]	S5
RIE_SO	2	20	3	0	0	Riemann problem for Euler equations with Shu-Osher sine wave hitting shock	[189]	S5
RIE_SOD	1	6	3	0	0	Sod's Riemann problem for Euler equations of a polytropic gas	[189]	X
ROD	4	10	1	0	0	Rod of solid explosive		S1
SALINE	4	20	2	2	0	Diffusion of drug in a saline solution through membrane	[312]	S5
SE_PULSE	2	23	1	0	0	Advection of semi ellipse pulse	[109]	S5
SETTLER	3	10	1	0	0	Solid dynamics within settling zone	[180]	S5
SH_FRONT	2	20	1	0	0	PDE with sharp front, exact solution known	[74]	X
SHEAR	4	33	3	0	0	Shear band formation	[235], [102], [284]	S1
SIN_GOR1	2	80	2	0	0	Sine-Gordon equation, exact kink-soliton solution	[279]	X
SIN_GOR2	1	80	2	0	0	Sine-Gordon equation, exact kink-kink-collision solution	[279]	X
SINGSTEP	2	130	1	0	0	Single-step reaction with diffusion	[2]	S1
SKIN_X	5	25	2	4	0	Transdermal diffusion	[284], [40]	ED
SKIN1	10	25	2	4	0	Transdermal diffusion	[284], [40]	E
SKIN10	10	25	2	4	0	Transdermal diffusion (STEP 0: First trial with all parameters)	[284], [40]	E
SKIN11	4	25	2	4	0	Transdermal diffusion (STEP 1: Data fitting with significant parameters)	[284], [40]	E
SKIN12	6	25	2	4	0	Transdermal diffusion (STEP 3: Experimental design)	[284], [40]	E
SKIN13	6	25	2	4	0	Transdermal diffusion (STEP 4: Evaluate confidence intervals at initial design)	[284], [40]	E

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
SKIN14	6	25	2	4	0	Transdermal diffusion (STEP 5: Evaluate confidence intervals at final design)	[284], [40]	E
SKIN15	6	25	2	4	0	Transdermal diffusion (STEP 5: Experimental design with weights)	[284], [40]	E
SKIN16	3	3	2	4	0	Transdermal diffusion (STEP 7: Identifiability at three optimal weights)	[284], [40]	E
SKIN2	8	25	3	6	0	Skin model with association kinetics	[40]	E
SKIN3	3	25	2	4	0	Skin model, in vitro experiment, with perfect sink	[40]	E
SKIN3_X	6	68	2	4	0	Skin model, in vitro experiment, with perfect sink	[40]	ED
SKIN4	3	56	2	4	0	Transdermal diffusion	[40]	S1
SKIN5	7	25	2	4	0	Transdermal diffusion	[284]	E
SLAB	3	36	3	0	0	Dwyer-Sanders flame propagation model	[242]	S5
SLAB_CTR	20	10	1	0	16	Temperature control of a slab	[14]	none
SOIL	3	80	2	0	0	Diffusion of water through soil, convection and dispersion	[339], [5], [284]	E
SOLID	2	14	1	0	0	Heating of solid sphere	[13]	S5
SOLITON	2	20	2	0	0	Kink soliton (Sine-Gordon equation)		S5
SORP_IS1	2	11	1	0	3	Reactive solute transport, convective-diffusive transport (Freundlich)	[145]	S5
SORP_IS2	2	11	1	0	0	Reactive solute transport, convective-diffusive transport (Langmuir)	[145]	S5
SORP_IS3	4	22	1	0	3	Reactive solute transport, convective-diffusive transport	[145]	S5
SORPTION	3	41	1	1	3	Transport equation (diffusion and sorption), ground water flow with contamination		E
SOUND	2	10	1	0	0	Sound in tissue	[284], [40]	S1
SPHERE	2	16	1	0	0	Heat conduction in sphere with exothermic chemical reaction	[277]	S5
STAR_NET	3	30	3	1	0	Parabolic star net		S1
STARTBED	2	81	1	0	0	Diffusion		E
STARTUP	3	30	11	0	0	Startup phase of an automobile catalytic converter	[85]	S1

(continued)

name	n	$\tilde{l}$	n_p	n_c	m_r	*background*	*ref*	*data*
STEPHAN	2	110	1	1	0	One-phase Stephan problem	[27]	X
STFFDET1	2	11	1	0	0	Stiffness detection	[88]	S5
STFFDET2	2	15	1	0	0	Stiffness detection	[88]	S5
STR_FISH	2	18	1	0	0	Stream fish tracked by mark-recapture technique		S5
T_DIFFUS	5	45	1	3	0	Transdermal diffusion through two membranes with transition layer		E
TELEGRPH	4	16	2	0	0	Telegraph equation	[278]	S5
TIME_OPT	6	10	1	0	5	Time-optimal heat distribution	[281]	E
TONGUE	3	20	1	0	0	Motion of glacier tongue	[63]	S5
TRAFFIC	1	80	1	0	0	Traffic flow along a highway	[151]	S5
TRAN_DEG	3	32	1	0	0	Saturation of ground water (Richards equation)	[338], [264]	S1
TRANSDER	3	45	1	3	0	Transdermal diffusion		E
TRANSMEM	5	45	1	3	0	Two membranes with transition area		E
TRAV_WAV	2	140	1	0	0	Traveling waves (Burger's equation, exact solution known)	[2]	X
TUBE	1	63	3	0	0	Tube with gas separated by membrane, shock distribution	[324]	S1
TUBE0	2	10	1	0	0	Zero-order reaction in a catalytic-walled tube	[53]	S5
TWO_POPS	3	40	2	0	0	Two populations	[332]	S5
VISCOUS	5	5	5	5	0	Variable viscosities with periodic boundary	[9]	S5
WATER	1	50	2	0	0	Flow of shallow water over a barrier	[309], [142]	S1
WAVE1	1	180	2	0	0	Hyperbolic wave equation (exact solution known)	[277]	S0
WAVE2	2	100	2	0	0	Wave equation in form of two hyperbolic equations		X
WAVE3	2	60	2	0	0	Hyperbolic wave equation	[353]	S5
WAVE4	3	50	2	0	0	Two waves traveling in opposite directions, semi-hyperbolic system	[346], [284]	S1

3.8 Partial Differential Algebraic Equations

Again we proceed from r data sets

$$(t_i, y_i^k)\ ,\ i = 1, \ldots, l_t,\ k = 1, \ldots, r\ \ ,$$

where l_t time values and $l = l_t r$ corresponding measurement values are defined together with l weights w_i^k. Some of the weights can become zero in cases when the corresponding measurement value is missing, if artificial data are needed, or if plots are to be generated for state variables for which experimental data do not exist. The subsequent table contains the actual number $\tilde{l} \leq l$ of terms taken into account in the final least squares formulation.

The system of partial differential algebraic equations under consideration is

$$\begin{aligned}
\dot{u}_1 &= F_1(p, u, u_x, u_{xx}, v, x, t)\ \ , \\
&\cdots \\
\dot{u}_{n_d} &= F_{n_d}(p, u, u_x, u_{xx}, v, x, t)\ \ , \\
0 &= F_{n_d+1}(p, u, u_x, u_{xx}, v, x, t)\ \ , \\
&\cdots \\
0 &= F_{n_d+n_a}(p, u, u_x, u_{xx}, v, x, t)\ \ ,
\end{aligned}$$

where $u_d = (u_1,\ \ldots,\ u_{n_d})^T$ and $u_a = (u_{n_d+1},\ \ldots,\ u_{n_d+n_a})^T$ are the differential and algebraic state variables, $u = (u_d, u_a)^T$. $v \in I\!R^{n_c}$ denotes the state variables belonging to the coupled system of ordinary differential and algebraic equations. To simplify the notation, flux functions are omitted.

Initial and boundary conditions may depend on the parameter vector to be estimated. Since the starting time is assumed to be zero initial values have the form

$$u(p, x, 0) = u_0(p, x)\ \ ,$$

where $u = (u_d, u_a)^T$ is the combined vector of all differential and algebraic state variables. For both end points x_L and x_R we allow Dirichlet or Neumann boundary conditions

$$\begin{aligned}
u(p, x_L, t) &= u^L(p, v, t)\ \ , \\
u(p, x_R, t) &= u^R(p, v, t)\ \ , \\
u_x(p, x_L, t) &= \hat{u}^L(p, v, t)\ \ , \\
u_x(p, x_R, t) &= \hat{u}^R(p, v, t)
\end{aligned}$$

for $0 < t \leq T$, where T is the final integration time, for example the last experimental time value t_{l_t}. They may depend on the coupled ordinary differential and algebraic state variables. We do not require the evaluation of all boundary functions. Instead, we omit some of them depending on the structure of the PDAE model, for example, whether second partial derivatives exist in the right-hand side or not. Moreover, arbitrary implicit boundary conditions can be formulated in form of coupled algebraic equations.

However, we must treat initial and boundary conditions with more care. We have to guarantee that at least the boundary and transition conditions satisfy the algebraic equations

$$\begin{aligned}
0 &= F_a(p, u(p, x_L, t), u_x(p, x_L, t), u_{xx}(p, x_L, t), v, x_L, t)\ \ , \\
0 &= F_a(p, u(p, x_R, t), u_x(p, x_R, t), u_{xx}(p, x_R, t), v, x_R, t)\ \ .
\end{aligned}$$

If initial conditions for discretized algebraic equations are violated, that is if equation

$$0 = F_a(p, u(p, x, 0), u_x(p, x, 0), u_{xx}(p, x, 0), v(p, 0), x, 0)$$

is inconsistent after inserting Dirichlet or Neumann boundary values and corresponding approximations for spatial derivatives, the corresponding system of nonlinear equations is solved internally proceeding from initial values given.

Each set of experimental data is assigned a spatial variable value $x_k \in [x_L, x_R]$, $k = 1, \ldots, r$, where r denotes the total number of measurement sets. Some or all of the x_k-values may coincide, if different measurement sets are available at the same local position. Since partial differential equations are discretized by the method of lines, the fitting points x_k are rounded to the nearest line.

The resulting parameter estimation problems is

$$\begin{aligned} \min \quad & \sum_{k=1}^{r} \sum_{i=1}^{l_t} (w_i^k (h_k(p, u(p, x_k, t_i), u_x(p, x_k, t_i), \\ & \qquad u_{xx}(p, x_k, t_i), v(p, t_i), t_i) \;-\; y_i^k))^2 \\ p \in I\!R^n : \quad & g_j(p) = 0 \;, \quad j = 1, ..., m_e \;, \\ & g_j(p) \geq 0 \;, \quad j = m_e + 1, ..., m_r \;, \\ & p_l \leq p \leq p_u \;, \end{aligned}$$

where $v(p,t)$ is the solution vector of an optional system of n_c coupled ordinary differential and algebraic equations similar to the previous section. It is assumed that all model functions $h_k(p, u, u_x, u_{xx}, v, t)$ and $g_j(p)$ are continuously differentiable subject to p for $k = 1, \ldots, r$ and $j = 1, \ldots, m_r$, and also the state variables and their spatial derivatives $u(p, x, t)$, $u_x(p, x, t)$, $u_{xx}(p, x, t)$, and $v(p, t)$.

Test problems with one-dimensional partial differential algebraic equations are listed in Table 7. Not listed are the number of integration areas, switching times, and structure of the boundary conditions. There are no equality constraints.

Table 7. Partial Differential Algebraic Equations

name	n	$\tilde{l}$	n_d	n_a	n_c	m_r	*background*	*ref*	*data*
2ND_DIR1	3	20	1	1	0	0	Second order Dirichlet problem	[182]	S5
2ND_DIR2	3	20	1	1	0	0	Second order inhomogeneous Dirichlet problem	[182]	S5
ACCRET_A	1	20	6	4	0	0	Thermal equilibrium curves in Keplerian accretion disks	[252]	S5
ACCRET_F	1	112	3	1	0	0	Thermal equilibrium curves in Keplerian accretion disks	[252]	S5
ACETYL_T	2	180	10	9	0	0	Tubular acetylene reactor, time-dependent formulation	[34]	S0
ACETYL_Z	2	20	10	1	0	9	Tubular acetylene reactor, space-dependent formulation	[34]	S1
BEAM1	3	99	2	2	0	0	Curved beam	[340], [328]	S5
BEAM2	3	90	2	2	0	0	Linked beams	[340] , [328]	S5
BIFURC1	3	180	2	0	2	0	Bifurcation with codimension 2 (Ginzburg-Landau equation)	[6]	S1
BIFURC2	2	300	2	0	2	0	Bifurcation with codimension 2 (Ginzburg-Landau equation), dense observation grid	[6]	S1

(continued)

name	n	$\tilde{l}$	n_d	n_a	n_c	m_r	*background*	*ref*	*data*
BVP_TRIV	1	10	1	1	0	0	Boundary value problem with known solution	[174]	X
CAPILL	2	13	1	1	0	0	Capillar filled with water under electric charge		S5
CHRESIGN	8	303	9	3	0	0	Transport effects for chromatographic resigns		S5
CNT_CUR2	4	80	3	1	0	0	Counter-current separation of fluid phase concentrations with phase equilibrium	[250]	S5
CO_OXYD	3	68	2	0	2	0	CO oxidation on Pt(110)	[20]	S5
CTFLOW	1	40	4	2	0	0	Two incompressible counter-current flows of binary liquid mixture with semi-permeable wall	[222]	S1
CUSP	2	40	3	0	3	0	Threshold-nerve impulse with cusp catastrophe	[366]	S5
ELA_TUBE	3	40	2	1	0	0	Waves propagating in a liquid-filled elastic tube (Korteweg-de Vries-Burgers equation)	[152]	X
ELDYN_A	3	20	4	2	0	0	Electrodynamic application with algebraic equations	[37]	S5
EW_WAVE	2	48	2	1	0	0	Wave propagation in media with nonlinear steepening and dispersion	[130]	X
FLAT_MEM	1	20	1	0	2	0	Concentration boundary layer over a flat membrane		E
FLOWDIFF	2	101	2	0	2	0	Flow system with diffusion	[3]	S5
HEAT_A	1	27	2	1	0	0	Heat equation, formulated with algebraic equation		S5
HEAT_F	2	27	2	1	0	0	Heat equation, formulated with algebraic equation and flux formulation		S5
HEAT_NLC	2	10	1	0	1	0	Heat equation with nonlinear boundary condition of Stefan-Boltzmann type	[331]	E
HEAT_NLD	2	10	1	0	1	1	Heat equation with nonlinear boundary condition of Stefan-Boltzmann type	[331]	none
HEAT_R	2	27	2	1	0	9	Heat equation with dynamical restrictions and algebraic equation		S5

(continued)

name	n	$\tilde{l}$	n_d	n_a	n_c	m_r	*background*	*ref*	*data*
HEAT_RAD	2	10	2	1	0	0	Heat conduction with radiation and forced convection	[115]	S5
KDV1	2	44	2	1	0	0	Korteweg-de-Vries equation, exact solution with one soliton	[279]	X
KDVE	1	17	2	1	0	0	Shallow water flow, balancing front sharpening and dispersion to produce solitons	[279]	S0
MC_DIST	4	63	4	2	4	0	Multi-component distillation	[222]	S5
MCFC1	20	20	19	2	0	0	MCFC fuel cell	[133],[62]	none
MCFC2	3	110	19	2	0	0	MCFC fuel cell	[133],[62]	S1
MCFC3	3	110	19	2	0	0	MCFC fuel cell	[133],[62]	S1
MCMC_0	5	70	19	2	9	0	MCMC fuel cell		S1
NET_3	10	100	2	1	2	0	Network with three beams and controlled Neumann knot	[110]	E
PAR_SINA	2	84	2	1	0	0	Parabolic PDE with inhomogeneous sinus-term	[262], [251]	X
PDAE4	2	6	2	1	0	0	Simple fourth order PDAE with exact solution		X
PLASMA	3	20	4	2	0	0	Space-time movement of ions and electrons	[197]	S5
PRESSURE	3	15	3	1	0	0	Pressure-driven flow in porous media	[191]	S5
SILICON	1	102	2	1	0	0	Diffusion in silicon		S5
TUNNEL	2	24	2	1	0	0	Tunnel furnace with heating and cooling section	[222]	S1
UNI_BEAM	2	10	5	3	0	0	Thin uniform cantilevered beam	[115]	S5
VIB_BEAM	6	6	3	1	0	0	Boundary control of transverse vibrations of a beam	[23]	E

4 Routines of the Modeling Language PCOMP

The model function describing, e.g., the right hand of a differential equation, boundary or initial values, constraints, and fitting criteria, are implemented in the modeling language PCOMP, see Dobmann, Liepelt, Schittkowski and Trassl [76], Dobmann, Liepelt, Schittkowski [75], or Liepelt and Schittkowski [184].

The PCOMP system consists of three modules that can be executed independently from each other. There are also some auxiliary routines, in particular an error routine called SYMERR to make error messages readable, and a routine with the name SYMPRP to read intermediate code generated by the parser.

(1) **Parser:**
The source code is analyzed and compiled into an intermediate code, which can then be processed by the other routines. The subroutine to be executed has the name SYMINP. The syntax of the code is described in the form of a formal grammar, see Appendix A. The parser was generated in C by the *yacc*-compiler-compiler of UNIX and then transformed into Fortran by hand. The following files are needed to link the parser:

PCOMP_P1.FOR	-	parser routines
PCOMP_P2.FOR	-	parser routines
PCOMP_EV.FOR	-	numerical evaluation of analytical expressions used
	-	in index or constant declarations
PCOMP_EX.FOR	-	external functions provided by the user
PCOMP_ER.FOR	-	error messages

To give an example, we list a possible implementation:

```
      parameter (lrsym=15000, lisym=15000)
      double precision rsym(lrsym)
      integer isym(lisym), larsym, laisym, ierr, lrow
      integer nvar,nfunc
      integer dbglev
      open(2,file='pcomp.fun', status='UNKNOWN')
      open(3,file='pcomp.sym', status='UNKNOWN')
      open(4,file='debug.fil',status='UNKNOWN')
      dbglev=3
      call SYMINP(2,3,rsym,lrsym,isym,lisym,larsym,laisym,ierr,lrow,
     F            1,nvar,nfunc,4,dbglev)
      if (ierr.gt.0) goto 900
      goto 9999
  900 call SYMERR(ierr,lrow)
 9999 continue
      close(2)
      close(3)
      close(4)
      stop
      end
```

(2) **Evaluation of Function, Gradient and Hessian Matrix:**
Proceeding from an intermediate code generated by SYMINP, the function and derivative values are computed within subroutines called SYMFUN, SYMGRA and SYMHES. They can be linked to any user program as required by the underlying application. First and second derivatives are computed by forward accumulation. The following program files are available and must be linked to the code provided by the user:

PCOMP_S.FOR	-	evaluation of function, gradient and Hessian matrix
PCOMP_EV.FOR	-	evaluation of expressions from given postfix notation
PCOMP_EX.FOR	-	external functions provided by the user
PCOMP_ER.FOR	-	error messages

By the next example, we illustrate a possible implementation of the routines for evaluating function and derivative values. We assume that the symbol file *pcomp.sym* contains the intermediate code of one function with two variables.

```
      implicit double precision(a-h,o-z)
      parameter (nmax=30, mmax=20, lrsym=50000, lisym=10000)
      dimension x(nmax), f(mmax), df(mmax,nmax),
     /          ddf(mmax,nmax*nmax),irsym(lrsym),rsym(lisym)
      logical act(mmax)
      integer nvar,nfunc
      integer dfx(nmax),dfxlen
      open(3,file='pcomp.sym',status='UNKNOWN')
      n=2
      m=1
      x(1)=1.0
      x(2)=-1.2
      act(1)=.true.
      dfx(1)=1
      dfx(2)=2
      dfxlen=2
      call SYMPRP(3,rsym,lrsym,isym,lisym,larsym,laisym,ierr,2,
     /            nvar,nfunc)
      if (ierr.gt.0) goto 900
      call SYMFUN(x,n,f,m,act,rsym,lrsym,isym,lisym,dfx,dfxlen,ierr)
      if (ierr.gt.0) goto 900
      call SYMGRA(x,n,f,m,df,mmax,act,rsym,lrsym,isym,lisym,dfx,dfxlen,
     /            ierr)
      if (ierr.gt.0) goto 900
      call SYMHES(x,n,f,m,df,ddf,mmax,act,rsym,lrsym,isym,lisym,dfx,
     /            dfxlen,ierr)
      if (ierr.gt.0) goto 900
      write(*,*) f(1),df(1,1),df(1,2)
      write(*,*) ddf(1,1),ddf(1,2),ddf(1,3),ddf(1,4)
      goto 9999
  900 call SYMERR(ierr,0)
 9999 continue
      close(3)
      stop
      end
```

(3) **Generation of Fortran Code:**
Proceeding from an intermediate code generated by SYMINP, Fortran subroutines for function and gradient evaluation are generated. They can be compiled and linked separately from the PCOMP system. Gradients are computed by reverse accumulation. There are two files to be linked to the user code, and the error routine as before:

PCOMP_G1.FOR - routines to generate Fortran code
PCOMP_G2.FOR - routines to generate Fortran code
PCOMP_ER.FOR - error messages

```
      parameter (lrsym=15000, lisym=15000)
      double precision rsym(lrsym)
      integer isym(lisym), larsym, laisym, ierr, lrow,nvar,nfunc
      open(3,file='pcomp.sym', status='UNKNOWN')
      open(4,file='pcomp.for', status='UNKNOWN')
      call SYMPRP(3,rsym,lrsym,isym,lisym,larsym,laisym,ierr,1,
     /            nvar,nfunc)
      if (ierr.gt.0) goto 900
      call SYMFOR(4,rsym,lrsym,isym,lisym,ierr)
      if (ierr.gt.0) goto 900
      goto 9999
  900 call SYMERR(ierr,0)
 9999 continue
      close(3)
      close(4)
      stop
      end
```

Documentation of all subroutines within the files mentioned is given by Dobmann (1993), Liepelt (1990) and Trassl (1993) together with some additional information on the data structures. In the remainder of this section, we describe only the use of subroutines that can be called from a user program.

Subroutine SYMINP:

- **Purpose:**
  The subroutine compiles symbolically defined nonlinear functions and generates an intermediate code.
- **Calling sequence:**

SYMINP (	INPUT,	SYMFIL,	WA,	LWA,	IWA,
	LIWA,	UWA,	UIWA,	IERR,	LNUM,
	MODE,	NVAR,	NFUNC,	DEBFIL,	LEVDEB)

- **Parameters:**

INPUT	-	When calling SYMINP, the integer value of INPUT is the number of the file that contains the program text.
SYMFIL	-	An integer identifying the output file number to which the intermediate code is to be written. If SYMFIL is set to zero when calling SYMINP, then only the working arrays are filled with the intermediate program code.
WA(LWA)	-	Double precision working array of length LWA used internally to store and process data. When leaving SYMINP, WA contains the generated intermediate code in its first UWA positions.

LWA	-	Length of the working array WA. LWA must be sufficiently large depending on the code size.
IWA(LIWA)	-	Integer working array of length LIWA. On return, IWA contains the integer part of the intermediate code in its first UIWA positions.
LIWA	-	Length of the working array IWA. LIWA must be sufficiently large depending on the code size.
UWA,UIWA	-	Storage actually needed for the intermediate code in the form of integers.
IERR	-	On return, IERR shows the termination reason of SYMINP: IERR = 0 : Successful termination. IERR > 0 : There is a syntax error in the input file. In the latter case, call SYMERR for more information.
LNUM	-	In case of unsuccessful termination, LNUM contains the line number where the error was detected.
MODE	-	MODE = 0 : WA is supposed to supply sufficient working space to store function values.
	-	MODE = 1 : WA is supposed to supply sufficient working space to store function values and corresponding first derivatives.
	-	MODE = 2 : WA is supposed to supply sufficient working space to store function values and corresponding first and second derivatives.
NVAR	-	On return, NVAR contains the number of variables of the INPUT file
NFUNC	-	On return, NFUNC contains the number of functions of the INPUT file
DEBFIL	-	An integer identifying the output file number to which debug information is to be written, if required the parameter LEVDEB.
LEVDEB	-	When calling SYMINP, LEVDEB has to contain the desired debugging level: LEVDEB = 0 : No debugging information generated. LEVDEB = 1 : Debug information generated by scanner. LEVDEB = 2 : Debug information generated by parser. LEVDEB = 3 : Full debug information generated.

Subroutine SYMERR:

- **Purpose:**
  Proceeding from an error code IERR (> 0) and, if available from a SYMINP call, a line number, SYMERR generates an output message on the standard device.
- **Calling sequence:**
  SYMERR (LNUM, IERR)
- **Parameters:**

LNUM	-	When calling SYMERR after a SYMINP execution, LNUM has to contain the corresponding line number value as determined by SYMINP.
IERR	-	The numerical value of the termination reason is to be inserted when calling SYMERR.

Subroutine SYMPRP:

- **Purpose:**
  The subroutine reads intermediate code from a file generated by a SYMINP call and fills two working arrays with the code for further processing within subroutines SYMFUN, SYMGRA and SYMFOR.

- **Calling sequence:**

```
SYMPRP (  SYMFIL,  WA,     LWA,   IWA,    LIWA,
          UWA,     UIWA,   IERR,  MODE,   NVAR,
          NFUNC)
```

- **Parameters:**

SYMFIL	-	An integer identifying the input file number, which contains the intermediate code generated by SYMINP.
WA(LWA)	-	Double precision working array of length LWA that contains the intermediate code in its first UWA positions when leaving SYMPRP.
LWA	-	Length of the working array WA. LWA must be at least UWA as determined by SYMINP.
IWA(LIWA)	-	Integer working array of length LIWA. On return, IWA contains the integer part of the intermediate code in its first UIWA positions.
LIWA	-	Length of the working array IWA. LIWA must be at least UIWA as determined by SYMINP.
UWA,UIWA	-	Storage actually needed for the intermediate code in WA and IWA.
IERR	-	On return, IERR shows the termination reason of SYMPRP: IERR = 0 : Successful termination. IERR > 0 : There is an error in the input file. In the latter case, call SYMERR for more information.
MODE	-	MODE = 0 : WA is tested for sufficient space to store function values.
	-	MODE = 1 : WA is tested for sufficient space to store function values and corresponding first derivatives.
	-	MODE = 2 : WA is tested for sufficient space to store function values and corresponding first and second derivatives.
NVAR	-	On return, NVAR contains the number of variables of the INPUT file
NFUNC	-	On return, NFUNC contains the number of functions of the INPUT file

Subroutine SYMFUN:

- **Purpose:**
  The intermediate code is passed from a SYMINP call to SYMFUN in form of a real and an integer working array. Given any variable vector x, the subroutine computes the corresponding function values $f_i(x)$. The functions that are to be evaluated by SYMFUN must be specified by a logical array.

- **Calling sequence:**

```
SYMFUN (  X,       N,     F,    M,      ACTIVE,
          WA,      LWA,   IWA,  LIWA,   DFX,
          DFXLEN,  IERR)
```

- **Parameters:**

X(N)	-	Double precision array of length N that contains the variable values for which functions are to be evaluated.
N	-	Dimension, i.e. number of variables.
F(M)	-	Double precision array of length M to pass the function values computed by SYMFUN to the user program.
M	-	Total number of functions on the input file.
ACTIVE(M)	-	Logical array of length M that determines the functions to be evaluated. ACTIVE must be set by the user when calling SYMFUN: ACTIVE(J)=.TRUE. : Compute function $g_j(x)$. ACTIVE(J)=.FALSE. : Do not compute $g_j(x)$.
WA(LWA)	-	Double precision working array of length LWA that contains the intermediate code in its first UWA positions.
LWA	-	Length of the working array WA. LWA must be at least UWA as determined by SYMINP.
IWA(LIWA)	-	Integer working array of length LIWA. IWA contains the integer part of the intermediate code in its first UIWA positions.
LIWA	-	Length of the working array IWA. LIWA must be at least UIWA as determined by SYMINP.
DFX	-	Array for the evaluating function EVAL.
DFXLEN	-	Length of the array DFX.
IERR	-	On return, IERR shows the termination reason of SYMFUN: IERR = 0 : Successful termination. IERR > 0 : There is an error in the input file. In the latter case, call SYMERR for more information.

Subroutine SYMGRA:

- **Purpose:**
  The intermediate code is passed from a SYMINP call to SYMGRA in the form of a real and an integer working array. Given a variable vector x, the subroutine computes the corresponding function and gradient values $f_i(x)$ and $\nabla f_i(x)$. The functions and gradients that are to be evaluated by SYMGRA must be specified by a logical array.

- **Calling sequence:**

```
SYMGRA (  X,      N,       F,       M,     DF,
          MMAX,   ACTIVE,  WA,      LWA,   IWA,
          LIWA,   DFX,     DFXLEN,  IERR   )
```

- **Parameters:**

X(N)	-	Double precision array of length N that contains the variable values for which functions and gradients are to be evaluated.
N	-	Dimension, i.e. number of variables.
F(M)	-	Double precision array of length M to pass the function values computed by SYMGRA to the user program.

M	-	Total number of functions on the input file.
DF(MMAX,N)	-	Two-dimensional double precision array to take over the gradients computed by SYMGRA. The row dimension must be MMAX in the driving routine.
MMAX	-	Row dimension of DF. MMAX must not be smaller than M.
ACTIVE(M)	-	Logical array of length M that determines the functions and gradients to be evaluated. ACTIVE must be set by the user when calling SYMGRA: ACTIVE(J)=.TRUE. : Compute function $g_j(x)$. ACTIVE(J)=.FALSE. : Do not compute $g_j(x)$.
WA(LWA)	-	Double precision working array of length LWA that contains the intermediate code in its first UWA positions.
LWA	-	Length of the working array WA. LWA must be at least UWA as determined by SYMINP.
IWA(LIWA)	-	Integer working array of length LIWA. IWA contains the integer part of the intermediate code in its first UIWA positions.
LIWA	-	Length of the working array IWA. LIWA must be at least UIWA as determined by SYMINP.
DFX(DFXLEN)	-	INTEGER-array of length DFXLEN that contains the numbers of the variables for which the first derivatives are to be evaluated.
DFXLEN	-	Lenght of the vector DFX.
IERR	-	On return, IERR shows the termination reason of SYMGRA: IERR = 0 : Successful termination. IERR > 0 : There is an error in the input file. Call SYMERR for more information.

Subroutine SYMHES:

- **Purpose:**
  The intermediate code is passed from a SYMINP call to SYMHES in the form of a real and an integer working array. Given a variable vector x, the subroutine computes the corresponding function and derivative values $f_i(x)$, $\nabla f_i(x)$ and $\nabla^2 f_i(x)$. The functions, gradients and Hessian matrices that are to be evaluated by SYMHES must be specified by a logical array.

- **Calling sequence:**

```
SYMHES (  X,     N,      F,       M,       DF,
          DDF,   MMAX,   ACTIVE,  WA,      LWA,
          IWA,   LIWA,   DFX,     DFXLEN,  IERR )
```

- **Parameters:**

X(N)	-	Double precision array of length N that contains the variable values for which functions and gradients are to be evaluated.
N	-	Dimension, i.e. number of variables.
F(M)	-	Double precision array of length M to pass the function values computed by SYMGRA to the user program.
M	-	Total number of functions on the input file.

DF(MMAX,N)	-	Two-dimensional double precision array to take over the gradients computed by SYMGRA. The row dimension must be MMAX in the driving routine.
DDF(MMAX,N∗N)	-	Two-dimensional double precision array to i take over the values of the Hessian matrix computed by SYMHES. The row dimension must be MMAX in the driving routine.
MMAX	-	Row dimension of DF. MMAX must not be smaller than M.
ACTIVE(M)	-	Logical array of length M that determines the functions and gradients to be evaluated. ACTIVE must be set by the user when calling SYMGRA: ACTIVE(J)=.TRUE. : Compute function $g_j(x)$. ACTIVE(J)=.FALSE. : Do not compute $g_j(x)$.
WA(LWA)	-	Double precision working array of length LWA that contains the intermediate code in its first UWA positions.
LWA	-	Length of the working array WA. LWA must be at least UWA as determined by SYMINP.
IWA(LIWA)	-	Integer working array of length LIWA. IWA contains the integer part of the intermediate code in its first UIWA positions.
LIWA	-	Length of the working array IWA. LIWA must be at least UIWA as determined by SYMINP.
DFX(DFXLEN)	-	INTEGER-array of length DFXLEN that contains the numbers of the variables for which the first and second derivatives are to be evaluated.
DFXLEN	-	Lenght of the vector DFX.
IERR	-	On return, IERR shows the termination reason of SYMHES: IERR = 0 : Successful termination. IERR > 0 : There is an error in the input file. Call SYMERR for more information.

Subroutine SYMFOR:

- **Purpose:**
  The intermediate code is passed from a SYMINP call to SYMFOR in the form of a real and an integer working array. Then SYMFOR generates two subroutines for function and gradient evaluation on a given output file. The calling sequences of the generated subroutines are

  ```
  XFUN ( X,      N,  F,  M,  ACTIVE,
         IERR    )
  ```

  and

  ```
  XGRA ( X,      N,       F,     M,  DF,
         MMAX,   ACTIVE,  IERR   )
  ```

  where the meaning of the parameters is the same as for SYMFUN and SYMGRA, respectively.

- **Calling sequence:**

  ```
  SYMFOR ( XFIL,  WA,  LWA,  IWA,  LIWA,
           IERR   )
  ```

- **Parameters:**

XFIL	-	An integer containing the number of the file on which the codes are to be written.
WA(LWA)	-	Double precision working array of length LWA that contains the intermediate code in its first UWA positions when calling SYMFOR and additional storage that is required by SYMFOR.
LWA	-	Length of the working array WA.
IWA(LIWA)	-	Integer working array of length LIWA. IWA contains the integer part of the intermediate code in its first UIWA positions when calling SYMFOR, and is needed for additional working space.
LIWA	-	Length of the working array IWA.
IERR	-	On return, IERR shows the termination reason of SYMFOR: IERR = 0 : Successful termination. IERR > 0 : There is an error in the input file. in hte latter case, call SYMERR for more information.

5 The Data Format

5.1 Parameter Estimation in Explicit Model Functions, Steady State Systems, Ordinary Differential Equations and Differential Algebraic Systems

The test problem set comes with all data files generated by **EASY-FIT**^{ModelDesign} , for parameter estimation, optimal control, and optimal experimental design. We describe the format of the input data to allow quick implementation of interfaces also for own applications.

A file named <name>.DAT contains parameter estimation data, some problem information and optimization data in formatted form, where *name* stands for one of the 1.300 test problem identifiers. The first 6 columns may contain an arbitrary string to identify the corresponding input row, if allowed by the format.

1	a80	FILE	Path to result directory.
2	a6,4x,i5	MODEL	Problem name passed to subroutine SYSFUN for identifying data fitting model structure: 1 - explicit model function 2 - Laplace formulation of model function 3 - steady-state system of equations 4 - system of ordinary differential equations 5 - system of differential algebraic equations
3	a70	INFO 1	Long information string.
4	a20	INFO 2	Information string.
5	a20	INFO 3	Information string.
6	a20	INFO 4	Information string.
7	a20	INFO 5	Information string.
8	a10	INFO 6	Information string.
9	a10	INFO 7	Information string.

Line	Format	Name	Description
10	a6,4x,2i5	NPAR	Number of parameters to be optimized and any related quantity, e.g., number of break points.
11	a6,4x,i5	NRES	Total number of constraints without bounds.
12	a6,4x,i5	NEQU	Number of equality constraints.
13	a6,4x,2g20.4	RT,RC	Formatted input of NRES rows each containing two real numbers identifying the experimental time and concentration parameters for which a constraint is to be supplied. The order is arbitrary, but first the equality and subsequently the inequality constraints are to be defined. The data are rounded to the nearest actual time and concentration value.
14	a6,4x,i5	NODE	Number of differential equations. NODE can be zero if no differential equation is defined.
15	a6,4x,i5	NCONC	Number of concentration values. NCONC must be -1 or higher.
16	a6,4x,2i5	NTIME, MPLOT	Number of time points, must be greater than 0. Logarithmic scaling of x-axis (MPLOT=1) or not (MPLOT=0).
17	a6,4x,i5	NMEAS	Number of measurement sets, i.e., of model functions with respect to which measurements are supplied. NMEAS is the dimension of the fitting function.
18	a6,4x,i5	NPLOT	Number of plot points, for example.
19	a6,4x,i5	NOUT	Output flag.
20	a6,4x,4i5	METHOD,	Choice of numerical algorithm and auxiliary information, e.g., shooting index, data fitting norm, or gradient approximation.
21	a6,4x,i5	OPTP1	Optimization parameter, e.g., maximum number of iterations.
22	a6,4x,i5	OPTP2	Optimization parameter.
23	a6,4x,i5	OPTP3	Optimization parameter.
24	a6,4x,g10.4	OPTE1	Optimization tolerance, e.g., for termination.

Line	Format	Name	Description
25	a6,4x,g10.4	OPTE2	Optimization tolerance.
26	a6,4x,g10.4	OPTE3	Optimization tolerance.
27	a6,4x,i5	ODEP1	Parameter for ODE solver.
28	a6,4x,5i5	ODEP2	Parameter for ODE solver.
29	a6,4x,i5	ODEP3	Parameter for ODE solver
30	a6,4x,5i5	ODEP4	Tolerance for ODE solver.
31	a6,4x,g10.4	ODEE1	Tolerance for ODE solver.
32	a6,4x,g10.4	ODEE2	Tolerance for ODE solver.
33	a6,4x,g10.4	ODEE3	Tolerance for ODE solver.
34	a6,4x,3g20.8	XL,X,XU	Formatted input of NPAR rows each containing three real numbers for - lower bound for estimated parameter - starting value for estimated parameter - upper bound for estimated parameter
35	a6,4x,i5	SCALE	Scaling information, for example
36	*		In case of NCONC>0, unformatted input of NTIME*NCONC rows for $j = 1$ to NCONC and $i = 1$ to NTIME (in this order) with the following data: t_i - i-th measurement time, not smaller than zero c_j - j-th concentration value y_{ij}^k, w_{ij}^k - measured data, i.e., experimental output, and individual weight factor for measurement with number k, $k = 1, \ldots$,NMEAS Otherwise these lines contain the following data in unformatted form for $i = 1$ to NTIME : t_i - i-th measurement time, not smaller than zero

Line	Format	Name	Description
			y_i^k, w_i^k - measured data, i.e., experimental output, and individual weight factor for measurement with number k, $k = 1, \ldots$,NMEAS
37	a6,4x,i5	AUXP1	Auxiliary parameter, e.g., output flag for computing consistent initial values.
38	a6,4x,i5	AUXP2	Auxiliary parameter.
39	a6,4x,g10.4	AUXE1	Auxiliary tolerance, e.g., termination tolerance for solving consistent initial values.
40	a6,4x,i5	NBPC	Number of constant break points where integration is to be restarted with initial tolerances.
41	*		Unformatted input of NBPC rows each containing one time value in increasing order that represents a break point of the right-hand side of an ODE/DAE.

Example 5.1. (DFE1) *The first one consists of an explicit model function given in the form*

$$h(p,t) = a(\exp(-\beta t) + 0.1)\exp(-a_1 t) + b\exp(-b_1 t) + c\exp(-c_1 t)\sin(dt) \quad . \tag{12}$$

The input file must contain then the following information:

```
C:\EASYFIT\PROBLEMS\DFE1
DFE1          1
Test Problem DFE1
Demo
Artificial Data
10.2.1994
Schittkowski
x
t
NPAR          8    0
NRES          0
NEQU          0
NODE          0
NCONC         0
NTIME        10    0
NMEAS         1
NPLOT        50
NOUT          1
METHOD        1
OPTP1       100
OPTP2         8
OPTP3         2
OPTE1        1.0E-10
OPTE2           1.0
```

```
OPTE3            0.0
ODEP1          0    0    0    0    0
ODEP2          0
ODEP3          0
ODEP4          0
ODEE1            0.0
ODEE2            0.0
ODEE3            0.0
a               0.0             1.0         1000.0
bt            1.0E-5            0.1         1000.0
a1            0.0001           0.001        1000.0
b              0.001             5.0        10000.0
b1            1.0E-5           100.0        10000.0
c                0.0             0.0         1000.0
c1            1.0E-5            0.1         1000.0
d                0.0         0.00001        10000.0
SCALE          1
     0.0     25.00     1.0
     3.0     15.51     1.0
     6.0     16.08     1.0
     9.0     16.11     1.0
    12.0     15.48     1.0
    15.0     14.24     1.0
    18.0     12.50     1.0
    21.0     10.41     1.0
    22.5     9.299     1.0
    24.0     8.162     1.0
NLPIP =        0
NLPMI =        0
NLPAC =          0.0
NBPC           0
```

The corresponding PCOMP file is:

```
C
C-------------------------------------------------------
C
C     Problem:  DFE1
C
C     Date:     08.03.94
C
C-------------------------------------------------------
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (x)
C     2. concentration variable, if exists (c)
C     3. time variable (t)
C
*     VARIABLE
      a, bt, a1, b, b1, c, c1, d, t
C
C-------------------------------------------------------
C
C   - Fitting criteria:
```

```
C
*     FUNCTION y
      y = a*(exp(-bt*t) + 0.1)*exp(-a1*t) + b*exp(-b1*t)
     /    + c*exp(-c1*t)*sin(d*t)
C
C-------------------------------------------------------
C
C   - Constraints (if exist):
C
C*     FUNCTION G
C      G = ...
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

Example 5.2. (SE) *The second example describes a single differential equation with three concentration values,*

$$\dot{y} = k_1(r - y)(c - y) - k_2 y \quad . \tag{13}$$

Here the coefficients k_1, k_2, and r are to be estimated. The data file is setup in the following way:

```
C:\EASYFIT\PROBLEMS\SE
SE            4
Test Problem SE
Demo
Artificial Data
10.2.1994
Schittkowski
c
t
NPAR          3    0
NRES          0
NEQU          0
NODE          1
NCONC         3
NTIME        14    0
NMEAS         1
NPLOT        50
NOUT          1
METHOD        1
OPTP1       100
OPTP2         8
OPTP3         2
OPTE1         1.0E-9
OPTE2            1.0
OPTE3            0.0
ODEP1        11
ODEP2         1    0    0    0    0
ODEP3         6
```

```
ODEP4          0
ODEE1          1.0E-9
ODEE2          1.0E-9
ODEE3          1.0E-6
Kp            1.0E-4          1.0E-4            10.0
Km            1.0E-6          1.0E-5            0.01
R0             100.0           150.0            400.0
SCALE          1
   1.0     100.0    1.39    1.0
   3.0     100.0    4.06    1.0
   5.0     100.0    5.24    1.0
  10.0     100.0    11.6    1.0
  15.0     100.0    11.2    1.0
  20.0     100.0    17.4    1.0
  25.0     100.0     0.0    0.0
  30.0     100.0    20.6    1.0
  35.0     100.0    25.1    1.0
  40.0     100.0    25.3    1.0
  50.0     100.0    26.9    1.0
  60.0     100.0     0.0    0.0
  90.0     100.0    34.1    1.0
 105.0     100.0     0.0    0.0
  1.0     1000.0    16.5    1.0
  3.0     1000.0    39.1    1.0
  5.0     1000.0    57.4    1.0
  10.0    1000.0    97.3    1.0
  15.0    1000.0    118.5   1.0
  20.0    1000.0    139.6   1.0
  25.0    1000.0    159.8   1.0
  30.0    1000.0    173.2   1.0
  35.0    1000.0    180.3   1.0
  40.0    1000.0    189.9   1.0
  50.0    1000.0    200.4   1.0
  60.0    1000.0    209.3   1.0
  90.0    1000.0    202.4   1.0
 105.0    1000.0    213.1   1.0
   1.0    5000.0    72.9    1.0
   3.0    5000.0    153.8   1.0
   5.0    5000.0    189.4   1.0
  10.0    5000.0    239.7   1.0
  15.0    5000.0    250.6   1.0
  20.0    5000.0    234.1   1.0
  25.0    5000.0    255.1   1.0
  30.0    5000.0    254.7   1.0
  35.0    5000.0    266.2   1.0
  40.0    5000.0    270.3   1.0
  50.0    5000.0      0.0   0.0
  60.0    5000.0      0.0   0.0
  90.0    5000.0    237.7   1.0
 105.0    5000.0    250.8   1.0
NLPIP =       0
NLPMI =       0
NLPAC =         0.0
```

```
         NBPC          0
```

The corresponding PCOMP file is:

```
C
C-------------------------------------------------------
C
C     Problem:    SE
C
C     Date:       26.05.1994
C
C-------------------------------------------------------
C
*     REAL CONSTANT
      R=1.987
C
C-------------------------------------------------------
C
*     VARIABLE
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (x)
C     2. variables identifying solution of ordinary
C        differential equations (y)
C     3. concentration variable, if exists (c)
C     4. time variable (t)
C
*     VARIABLE
      Ep, A0, M, TEMP, T
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     ordinary differential equations:
C
*     FUNCTION DM
      DM = -A0*EXP(-Ep/(R*TEMP))*M
C
C-------------------------------------------------------
C
C   - Initial values for solving systems of ordinary
C     differential equations (t=0):
C
*     FUNCTION M0
      M0=1.0
C
C-------------------------------------------------------
C
C   - Fitting criterion:
C
*     FUNCTION FIT
      FIT=M
C
*     FUNCTION FIT2
```

```
      FIT2=2*M
C
C-------------------------------------------------------
C
C   - Constraints (if exist):
C
C*     FUNCTION G1
C      G1 = ...
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

Example 5.3. (VDPOL) *The input of data and model functions in case of a differential algebraic system, is to be outlined by van der Pol's equation. The model function is*

$$\dot{y} = z \ , \quad \dot{z} = y - a(1 - y^2)z \ . \tag{14}$$

We choose the consistent initial values

$$y^0 = a \ , \quad z^0 = b/(a(1 - b^2))$$

and consider a and b as parameters to be estimated. The fitting criteria are the solutions y and z. The data input file VDPOL.DAT has the following structure:

```
C:\EASYFIT\PROBLEMS\VDPOL
VDPOL         5
van der Pol's equation, electrical circuit
Demo
Schittkowski
Simulation
-
-
t
NPAR          2    0
NRES          0
NEQU          0
NODE          2
NCONC         0
NTIME         5    0
NMEAS         2
NPLOT        50
NOUT          0
METHOD        1
OPTP1        40
OPTP2         8
OPTP3         2
OPTE1        1.0E-07
OPTE2        1.0E-01
OPTE3        1.0E+02
ODEP1         5
```

```
ODEP2         1    1    2    0    0
ODEP3         6
ODEP4         0
ODEE1        1.0E-09
ODEE2        1.0E-06
ODEE3        1.0E-04
a               5.0E-01      1.001       10.0
b                   1.5      2.001        5.0
SCALE        -1
   0.0   2.000   1.0    -6.667E-1   1.0
2.0E-1   1.858   1.0    -7.575E-1   1.0
4.0E-1   1.693   1.0    -9.069E-1   1.0
6.0E-1   1.485   1.0       -1.233   1.0
8.0E-1   1.084   1.0       -6.200   1.0
NLPIP         0
NLPMI        50
NLPAC        1.0E-10
NBPC          0
```

The corresponding PCOMP file VDPOL.FUN is:

```
C
C-------------------------------------------------------
C
C     Problem:    VDPOL
C
C     Date:       22.12.97
C
C-------------------------------------------------------
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (x)
C     2. variables identifying solution of ordinary
C        differential equations (y)
C     3. variables identifying solution of corresponding
C        algebraic equations (z)
C     4. concentration variable, if exists (c)
C     5. time variable (t)
C
*     VARIABLE
      a, b, y, z, t
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     ordinary differential equations:
C
*     FUNCTION y_t
      y_t = z
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     algebraic equations:
```

```
C
*     FUNCTION alg_equ
      alg_equ = y - a*(1 - y**2)*z
C
C-------------------------------------------------------
C
C   - Initial values for solving systems of ordinary
C     differential equations (t=0):
C
*     FUNCTION y_0
      y_0 = b
C
C-------------------------------------------------------
C
C   - Initial values for solving systems of algebraic
C     equations (t=0):
C
*     FUNCTION z_0
      z_0 = b/(a*(1 - b**2))
C
C-------------------------------------------------------
C
C   - Fitting criteria:
C
*     FUNCTION y_fit
      y_fit = y
C
*     FUNCTION z_fit
      z_fit = z
C
C-------------------------------------------------------
C
C   - Constraints (if exist):
C
C*     FUNCTION g1
C      g1 = ...
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

Example 5.4. (RECLIG10) *To illustrate the implementation of a steady state system, we consider the following example that is similar to a receptor-ligand binding study with one receptor and two ligands,*

$$\begin{array}{lcl} z_1(1+p_1z_2+p_2z_3)-p_3 & = & 0\ , \\ z_2(1+p_1z_1)-p_4 & = & 0\ , \\ z_3(1+p_2z_1)-t & = & 0\ . \end{array} \tag{15}$$

The system parameters are z_1, z_2 and z_3, and the parameters to be estimated, are p_1, p_2, p_3, and p_4. t is the independent model or time variable to be replaced by experimental data. The fitting criterion is $\overline{h}(p,z,t) = p_4 - z_2$ and we use the starting values $z_1^0 = p_3$, $z_2^0 = p_4$ and $z_3^0 = t$ for solving the system of

nonlinear equations. The subsequent input file shows the parameters, tolerances and measurement values used.

```
C:\EASYFIT\problems\RECLIG10
RECLIG10      2
Steady state system, receptor-ligand binding study
Demo
Schittkowski
Simulation
nMol
Null
log nMol
NPAR  =       4    0
NRES  =       0
NEQU  =       0
NODE  =       3
NCONC =       0
NTIME =      13    0
NMEAS =       1
NPLOT =      50    0
NOUT  =       0
METHOD=      01    0
OPTP1 =     200
OPTP2 =      20
OPTP3 =      02
OPTE1 =      1.0E-09
OPTE2 =      1.0E+00
OPTE3 =      1.0E+02
ODEP1 =       0
ODEP2 =       1    3    0    0    0
ODEP3 =       0
ODEP4 =       0
ODEE1 =   0.0
ODEE2 =   0.0
ODEE3 =   0.0
p1        0.0          1.0         10000.0
p2        0.0          1.0         10000.0
p3        0.0        100.0         10000.0
p4        0.0          2.0         10000.0
SCALE =       1
1.0      0.332   1.0
5.0      0.331   1.0
1.0E+1   0.331   1.0
5.0E+1   0.327   1.0
1.0E+2   0.321   1.0
5.0E+2   0.289   1.0
1.0E+3   0.250   1.0
5.0E+3   0.125   1.0
1.0E+4   0.077   1.0
5.0E+4   0.019   1.0
1.0E+5   0.010   1.0
5.0E+5   0.002   1.0
1.0E+6   0.001   1.0
```

```
        NLPIP         0
        NLPMI       100
        NLPAC        1.0E-11
        NDISCO=       0
```

The corresponding PCOMP file is:

```
C
C-------------------------------------------------------
C
C     Problem:     RECLIG10
C
C     Date:        22.12.97
C
C-------------------------------------------------------
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (p)
C     2. system variables (z)
C     3. time variable (t)
C
*     VARIABLE
      x1, x2, x3, x4, z1, z2, z3, t
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     nonlinear equations:
C
*     FUNCTION g1
      g1 = z1*(1 + x1*z2 + x2*z3) - x3
C
*     FUNCTION g2
      g2 = z2*(1 + x1*z1) - x4
C
*     FUNCTION g3
      g3 = z3*(1 + x2*z1) - t
C
C-------------------------------------------------------
C
C   - Initial values for solving systems of nonlinear
C     equations:
C
*     FUNCTION z1_0
      z1_0 = x3
C
*     FUNCTION z2_0
      z2_0 = x4
C
*     FUNCTION z3_0
      z3_0 = t
C
C-------------------------------------------------------
```

```
C
C   - Fitting criterion:
C
*     FUNCTION fit
      fit = x4 - z2
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

5.2 Parameter Estimation in Partial Differential and Partial Differential Algebraic Equations

A file named <name>.DAT contains parameter estimation data, some problem information and optimization data in formatted form, where *name* stands for one of the 1.300 test problem identifiers. The first 6 columns may contain an arbitrary string to identify the corresponding input row, if allowed by the format.

1	a80	FILE	Path to result directory.
2	a6,4x,i5	MODEL	Problem name passed to subroutine SYSFUN for identifying data fitting model structure: 1 - explicit model function 2 - Laplace formulation of model function 3 - steady-state system of equations 4 - system of ordinary differential equations 5 - system of differential algebraic equations
3	a70	INFO 1	Long information string.
4	a20	INFO 2	Information string.
5	a20	INFO 3	Information string.
6	a20	INFO 4	Information string.

Line	Format	Name	Description
7	a20	INFO 5	Information string.
8	a10	INFO 6	Information string.
9	a10	INFO 7	Information string.
10	a6,4x,2i5	NPAR	Number of parameters to be optimized and any related quantity, e.g., number of break points.
11	a6,4x,i5	NPDE	Number of PDEs.
12	a6,4x,i5	NPAE	Number of algebraic equations.
13	a6,4x,i5	NCPLO,NCPLA	Numbers of coupled ordinary differential and algebraic equations.
14	a6,4x,i5	ICPLO	Formatted input of NCPLO lines where each line contains the line number, i.e., the discretization point, where the ODE is coupled to the PDE. The line with number 1 denotes the left boundary of the integration area.
15	a6,4x,i5	ICPLA	Formatted input of NCPLA lines where each line contains the line number, i.e., the discretization point, where the algebraic equation is coupled to the PDE. The line with number 1 denotes the left boundary of the integration area.
16	a6,4x,i5	NCPB	Number of area boundaries, which is an even number.
17	a6,4x,i5	NRES	Number of constraints without bounds.
18	a6,4x,i5	NEQU	Number of equality constraints.
19	a6,4x, 2g20.4, i5	RT,RX,IX	Formatted input of NRES rows each containing two real numbers identifying the experimental time and spatial parameter values for which a constraint is to be supplied, and the corresponding line number. The order is arbitrary, but first the equality and subsequently the inequality constraints are to be defined. The data are rounded to the nearest actual time value.
20	a6,4x,2i5	NTIME,	Number of time points, must be greater than 0.

Line	Format	Name	Description
		MPLOT	Logarithmic scaling of x-axis (MPLOT=1) or not (MPLOT=0).
21	a6,4x,i5	NFIT	Number of fitting criteria.
22	a6,4x,i5	IFIT	Formatted input of NFIT lines where each line contains the line number, i.e., the discretization point, where a fit criterion is defined. The line with number 1 denotes the left boundary of the integration area.
23	a6,4x,i5	NPLOT	Number of plot points, for example.
24	a6,4x,i5	NOUT	Output flag.
25	a6,4x,i5	PDEP1	Parameter for PDE solver.
26	a6,4x,i5	PDEP2	Parameter for PDE solver.
27	a6,4x,i5	DEP3	Parameter for PDE solver.
28	a6,4x,i5	PDEP4	Parameter for PDE solver.
29	a6,4x,3i5	METHOD,	Choice of numerical algorithm and auxiliary information, e.g., data fitting norm, or gradient approximation.
30	a6,4x,i5	OPTP1	Optimization parameter, e.g., maximum number of iterations.
31	a6,4x,i5	OPTP2	Optimization parameter.
32	a6,4x,i5	OPTP3	Optimization parameter.
33	a6,4x,g10.4	OPTE1	Optimization tolerance, e.g., for termination.
34	a6,4x,g10.4	OPTE2	Optimization tolerance.
35	a6,4x,g10.4	OPTE3	Optimization tolerance.
36	a6,4x,i5	ODEP1	Parameter for ODE solver.
37	a6,4x,5i5	ODEP2	Parameter for ODE solver.
38	a6,4x,i5	ODEP3	Parameter for ODE solver
39	a6,4x,5i5	ODEP4	Tolerance for ODE solver.
40	a6,4x,g10.4	ODEE1	Tolerance for ODE solver.

Line	Format	Name	Description
41	a6,4x,g10.4	ODEE2	Tolerance for ODE solver.
42	a6,4x,g10.4	ODEE3	Tolerance for ODE solver.
43	a6,4x,g10.4	XSTART	Value of the spatial component at the leftmost boundary
44	a6,4x,g10.4, 5i5 10x,3i5		Formatted input of $NCPB/2 \ldots (NPDE+1)$ lines for - the name of the area, - the spatial size of the area, - the number of discretization points in the area, whereas the following NPDE lines contain for each PDE - status of left boundary condition, - status of right boundary condition, - spatial derivative approximation . The boundary status is: 0 - no boundary condition 1 - Dirichlet boundary condition 2 - Neumann boundary condition
45	a6,4x,3g20.8		Formatted input of NPAR lines each containing three real numbers for - lower bound for estimated parameter - starting value for estimated parameter - upper bound for estimated parameter
46 47	a6,4x,i5 *	SCALE	Scaling information, for example. Unformatted input of NTIME lines for $i = 1 \ldots NTIME$ with the following data: t_i - i-th measurement time, not smaller than zero y_i^k, w_i^k - measured data, i.e., experimental output, and individual weight factor for measurement number k, $k = 1, \ldots, NMEAS$.

Line	Format	Name	Description
48	a6,4x,i5	AUXP1	Auxiliary parameter, e.g., output flag for computing consistent initial values.
49	a6,4x,i5	AUXP2	Auxiliary parameter.
50	a6,4x,i5	AUXP3	Auxiliary parameter.
51	a6,4x,i5	AUXP4	Auxiliary parameter.
52	a6,4x,g10.4	AUXE1	Auxiliary tolerance.
53	a6,4x,i5	NBPC	Number of constant break points where integration is restarted with initial tolerances.
54	*		Unformatted input of NBPC rows each containing one time value in increasing order that represents a break point of the right-hand side of a PDE.

Example 5.5. (HEAT) *We consider a simple heat conduction model found in Schiesser [277], where Fourier's first law for heat conduction leads to the equation*

$$u_t = u_{xx} \tag{16}$$

defined for $0 < t \leq 0.5$ *and* $0 < x < 1$. *Boundary conditions are*

$$u(0,t) = u(1,t) = 0 \tag{17}$$

for $0 \leq t \leq 0.5$ *and the initial values are*

$$u(x,0) = \sin\left(\frac{\pi x}{L}\right) \tag{18}$$

for $0 < x < 1$ *and* $0 < L \leq 1$. *In addition, we are interested in the total amount of heat at the surface* $x = 0$

$$\dot{v} = -K \cdot \frac{\pi}{L} \cdot e^{-\frac{\pi^2}{L^2} \cdot t} \tag{19}$$

with initial heat

$$v_0 = \frac{K \cdot L}{\pi}$$

Function v *serves as our fitting criterion. Parameters to be estimated, are* L *and* K. *Measurements are simulated subject to* $L = 1$, $K = 2$ *at the spatial coordinates 0, 0.1, 0.2, 0.3, 0.4, and 0.5. The corresponding data file HEAT.DAT is the following one, where starting values are* $L = 2$ *and* $K = 3$.

```
C:\EASYFIT\PROBLEMS\HEAT
HEAT          6
Heat conduction
One compartment
Test
18.01.1995
deg
mm
```

```
min
NPAR  =       2    0
NPDE  =       1
NPAE  =       0
NCPL  =       1    0
ICPL  =       1
NCPB  =       2
NRES  =       0
NEQU  =       0
NTIME =       6    0
NFIT  =       1
IFIT  =       1
NPLOT =      20
NOUT  =       1
DQUPOI=       3
APRMET=       0
FLUX  =       0
APRFLX=          0.0
OPTMET=       1
OPTP1 =     100
OPTP2 =       8
OPTP3 =       2
OPTE1 =       1.0E-7
OPTE2 =       1.0E-1
OPTE3 =       1.0E+2
ODEP1 =       4
ODEP2 =       0
ODEP3 =       6
ODEP4 =       0
ODEE1 =       1.0E-5
ODEE2 =       1.0E-5
ODEE3 =       1.0E-5
XSTART=          0.0
COMP1            1.0   11
                 1    1    0
L                0.0            2.0            10.0
K                0.0            3.0            10.0
SCALE =       0
0.0    0.636619      100.0
0.1    0.237273      100.0
0.2    8.84335E-2    100.0
0.3    3.29598E-2    100.0
0.4    1.22844E-2    100.0
0.5    4.57849E-3    100.0
INTEG =       0
ORDER =       2
NLPIP =       0
NLPMI =       0
NLPAC =        0.0
NBPC  =       0
```

The corresponding PCOMP file HEAT.FUN is:

```
C
```

```
C-------------------------------------------------------
C
C     Problem:    HEAT
C
C     Date:       18.4.95
C
C-------------------------------------------------------
C
*     REAL CONSTANT
      pi = 3.1415926535
C
C-------------------------------------------------------
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (p)
C     2. variables identifying solution of partial
C        differential equations, first the differential ones,
C        then the algebraic variables (u)
C     3. variables identifying first spatial derivatives of
C        solution of partial differential equations (u_x)
C     4. variables identifying second spatial derivatives of
C        solution of partial differential equations (u_xx)
C     5. variables identifying solution of coupled
C        ordinary differential equations (v)
C     6. variables for flux functions, if available (one
C        variable for each function (f), then one for
C        each derivative (f_x). These names can be
C        inserted subsequently. Note that flux variables
C        must be declared for each PDE equation, if at all.
C     7. spatial variable (x)
C     8. time variable (t)
C
*     VARIABLE
      c, u, u_x, u_xx, x, t
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     partial differential equations (order: differential
C     followed by algebraic functions in first area, then
C     differential followed by algebraic functions in
C     second area, etc.):
C
*     FUNCTION u_t
      u_t = c*u_xx
C
C-------------------------------------------------------
C
C   - Initial values of system of partial differential
C     equations for t=0 (order: initial values for
C     differential followed by algebraic variables in
C     first area, then differential followed by algebraic
C     variables in second area, etc.):
```

```
C
*     FUNCTION u_0
      u_0 = c*sin(pi*x)
C
C-------------------------------------------------------
C
C   - Boundary and transition functions first at left
C     border of an area, then at right border of an
C     area, etc.:
C
*     FUNCTION u_left
      u_left = 0
C
*     FUNCTION u_right
      u_right = 0
C
C-------------------------------------------------------
C
C   - Fitting criteria, order defined by increasing
C     spatial positions:
C
*     FUNCTION fit
      fit = u
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

Example 5.6. (PDAE4) *To see how partial differential algebraic equations are modeled, we consider a very simple fourth-order partial differential equation obtained from successive differentiation of* $u(x,t) = ae^{-\pi^4 t}\sin(\pi x)$*,*

$$u_t = -au_{xxxx}$$

or, equivalently, two second-order differential algebraic equations

$$\begin{aligned} u_t &= -av_{xx}\ , \\ 0 &= v - u_{xx} \end{aligned}$$

defined for $0 \le x \le 1$ *and* $t \ge 0$*. Initial values are* $u(x,0) = \sin(\pi x)$ *and* $v(x,0) = -\pi^2\sin(\pi x)$ *and boundary values are* $u(0,t) = u(1,t) = v(0,t) = v(1,t) = 0$ *for all* $t \ge 0$*. Function* u *is a possible fitting criterion and* a *an unknown parameter to be estimated from experimental data. The data input, where experimental data of the exact solution are inserted, is*

```
problems\PDAE4
PDAE4         7
Simple fourth order PDAE with exact solution
Demo
Schittkowski
Exact
Null
x
```

```
t
NPAR  =     002  000  002
NPDE  =      02
NPAE  =      01
NCPL  =      00   00
NCPB  =   00002
NRES  =   00000
NEQU  =   00000
NTIME =    0006    0
NFIT  =   00001
FITNO =   00011        5.000000E-01
NPLOT =   00050
NOUT  =       0
DQUPOI=       5
APRMET=       2
FLUX  =       0
APRFLX=          0.0
METHOD=      01    2   01    0    0
OPTP1 =   00100  005
OPTP2 =   00030
OPTP3 =      02
OPTE1 =   1.0000E-08
OPTE2 =   1.0000E-02
OPTE3 =   2.0000E+00
ODEP1 =      04
ODEP2 =       0
ODEP3 =   00006
ODEP4 =   00000
ODEE1 =   1.0000E-08
ODEE2 =   1.0000E-10
ODEE3 =   1.0000E-02
XSTART=   0.0000E+00
u(x,t)    1.000E+00 0021
u(x,t)       01   01   00
v(x,t)       01   01   00
a                0.000000E+00        2.000000E+00        1.000000E+05
b                0.000000E+00        2.000000E+00        1.000000E+05
SCALE =      00
0.00000E+00  1.00000000E+00 1.00E+00
1.00000E-02  3.77535411E-01 1.00E+00
2.00000E-02  1.42532987E-01 1.00E+00
3.00000E-02  5.38112497E-02 1.00E+00
4.00000E-02  2.03156523E-02 1.00E+00
5.00000E-02  7.66987814E-03 1.00E+00
INTEG =       1
ORDER =       2
NLPIP =       0
NLPMI =     060
NLPAC =   1.0000E-10
NDISCO=     000
```

The corresponding model function file PDAE4.FUN is given as follows,

```
C
```

```
C-------------------------------------------------------
C
C     Problem:    PDAE4
C
C     Date:       16.12.97
C
C-------------------------------------------------------
C
*     REAL CONSTANT
      pi = 3.1415926535
C
C-------------------------------------------------------
C
C   - Independent variables in the following order:
C     1. parameters to be estimated (p)
C     2. variables identifying solution of partial
C        differential equations (u)
C     3. variables identifying first spatial derivatives of
C        solution of partial differential equations (ux)
C     4. variables identifying second spatial derivatives of
C        solution of partial differential equations (uxx)
C     5. variables identifying solution of coupled
C        ordinary differential equations (v)
C     6. variables for flux functions, if available (one
C        variable for each function (f), then one for
C        each derivative (f_x). These names can be
C        inserted subsequently. Note that flux variables
C        must be declared for each PDE equation, if at all.
C     7. spatial variable (x)
C     8. time variable (t)
C
*     VARIABLE
      a, b, u, v, u_x, v_x, u_xx, v_xx, x, t
C
C-------------------------------------------------------
C
C   - Functions defining the flux depending on u and u_x,
C     if available (order: functions in first area, func-
C     tions in second area, etc.):
C
C*     FUNCTION FLUX
C      FLUX = ...
C
C-------------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     partial differential equations (order: functions in
C     first area, functions in second area, etc.):
C
*     FUNCTION u_t
      u_t = -a*v_xx
C
*     FUNCTION alg_equ
```

```
      alg_equ = v - u_xx
C
C-----------------------------------------------------
C
C   - Initial values of system of partial differential
C     equations for t=0 (order: functions in
C     first area, functions in second area, etc.):
C
*     FUNCTION u_0
      u_0 = b*sin(pi*x)
C
*     FUNCTION v_0
      v_0 = -pi**2*b*sin(pi*x)
C
C-----------------------------------------------------
C
C   - Functions defining right-hand side of system of
C     coupled ordinary differential equations:
C
c*     FUNCTION VP
c      VP = ...
C
C-----------------------------------------------------
C
C   - Initial values of system of coupled ordinary
C     differential equations for t=0:
C
c*     FUNCTION V0
c      V0 = ...
C
C-----------------------------------------------------
C
C   - Boundary and transition functions first at left
C     borders of all areas, then at right borders of all
C     areas:
C
*     FUNCTION u_left
      u_left = 0
C
*     FUNCTION u_right
      u_right = 0
C
*     FUNCTION v_left
      v_left = 0
C
*     FUNCTION v_right
      v_right = 0
C
C-----------------------------------------------------
C
C   - Boundary and transition gradients w.r.t. spatial
C     variable first at left borders of all areas, then
C     at right borders of all areas:
```

```
C
c*     FUNCTION u_x_left
c      u_x_left = ...
C
c*     FUNCTION u_x_right
c      u_x_right = ...
C
C-------------------------------------------------------
C
C   - Fitting criteria:
C
*     FUNCTION fit_u
      fit_u = u
c      fit_u = exp(-t*pi**4)*sin(pi*x)
C
C-------------------------------------------------------
C
C   - Constraints (if exist):
C
C*     FUNCTION G1
C      G1 = ...
C
C-------------------------------------------------------
C
*     END
C
C-------------------------------------------------------
C
```

References

[1] Abel O., Helbig A., Marquardt W. (1997): *Optimization approaches to control-integrated design of industrial batch reactors*, Technical Report LPT-1997-20, Lehrstuhl für Prozesstechnik, RWTH Aachen

[2] Adjerid S., Flaherty J.E. (1986): *A moving finite element method with error estimation and refinement for one-dimensional time dependent partial differential equations*, SIAM Journal on Numerical Analysis, Vol. 23, 778-796

[3] Alonso A.A., Banga J.R., Balsa-Canto J.R. (2002): *Model reduction of complex food processes with applications in control and optimization*, in: Computational Techniques in Food Engineering, Editorial CIMNE-UPC, Universidad Politecnica de Cataluna, Barcelona, ISBN 84-95999-13-7, 155-169

[4] Anderson D.H. (1983): *Compartmental Modeling and Tracer Kinetics*, Lecture Notes in Biomathematics, Vol. 50, Springer, Berlin

[5] Andersson F., Olsson B. eds. (1985): *Lake Gådsjön. An Acid Forest Lake and its Catchment*, Ecological Bulletins, Vol. 37, Stockholm

[6] Argentine M., Coullet P. (1997): *Chaotic nucleation of metastable domains*, Physical Reviews E, Vol. 56, 2359-2362

[7] Ascher U.M., Mattheij R., Russel R. (1995): *Numerical Solution of Boundary Value Problems*, SIAM, Philadelphia

[8] Ascher U.M., Petzold L.R. (1998): *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations,* SIAM, Philadelphia

[9] Ascher U., Ruuth S., Wettin B. (1995): *Implicit-explicit methods for time-dependent partial differential equations*, SIAM Journal on Numerical Analysis, Vol. 32, 797-823

[10] Atkinson A.C., Bogacka B. (2002): *Compound and other optimum designs for systems of nonlinear differential equations arising in chemical kinetics*, Chemometrics and Intelligent Laboratory Systems, Vol. 61,17 - 33

[11] Baake E., Schloeder J.P. (1992): *Modelling the fast fluorescence rate of photosynthesis*, Bulletin of Mathematical Biology, Vol. 54, 999-1021

[12] Baily J.E. , Ollis D.F. (1986): *Biochemical Engineering Fundamentals*, McGraw Hill, New York

[13] Balsa-Canto E., Alonso A.A., Banga J.R. (2002): *A novel, efficient and reliable method for thermal process design and optimization. Part I: Theory*, Journal of Food Engineering, Vol. 52, 227-234

[14] Balsa-Canto E., Banga J.R., Alonso A.A., Vassiliadis V.S. (1998): *Optimal control of distributed processes using restricted second order information*, Report, Chemical Engineering Lab., CSIC, University of Vigo, Spain

[15] Balsa-Canto E., Banga J.R., Alonso A.A., Vassiliadis V.S. (2001): *Dynamic optimization of chemical and biochemical processes using restricted second-order information*, Computers and Chemical Engineering, Vol. 25, 539-546

[16] Baltes M., Schneider R., Sturm C., Reuss M. (1994): *Optimal experimental design for parameter estimation in unstructured growth models*, Biotechnical Progress, Vol. 10, 480-488

[17] Banga J.R., Versyck K.J., Van Impe J.F. (2002): *Computation of optimal identification experiments for nonlinear dynamic process models: a stochastic global optimization apporoach*, Journal of Industrial Engineering and Chemical Research, Vol. 41, 2425-2430

[18] Banks H.T., Crowley J.M., Kunisch K. (1983): *Cubic spline approximation techniques for parameter estimation in distributed systems*, IEEE Transactions on Automatic Control, Vol. AC-28, No. 7, 773-786

[19] Banks H.T, Kunisch K. (1989): *Estimation Techniques for Distributed Parameter Systems,* Birkhäuser, Boston, Basel, Berlin

[20] Bär M., Hegger R., Kantz H (1999): *Fitting partial differential equations to space-time-dynamics*, Physical Reviews E, Vol. 59, 337-342

[21] Bartholomew-Biggs M.C. (1995): *Implementing and using a FORTRAN 90 version of a subroutine for non-linear least squares calculations*, Report, NOC, Hatfield

[22] Bauer I., Bock H.G., Körkel S., Schlöder J. (2000): *Numerical methods for optimum experimental design in DAE systems*, Journal of Computational and Applied Mathematics, Vol. 120, 1-25

[23] Bazeze A., Bruch J.C., Sloss J.M. (1999): *Numerical solution of the optimal boundary control of transverse vibrations of a beam*, Numerical Methods for Partial Differential Equations, Vol. 15, No. 5, 558-568

[24] Beck J.V., Arnold K.J. (1977): *Parameter Estimation in Engineering and Science*, John Wiley, New York

[25] Belohlav Z., Zamostny P., Kluson P., Volf J. (1997): *Application of a random-search algorithm for regression analysis of catalytic hydrogenizations*, Canadian Journal of Chemical Engineering, Vol. 75, 735-742

[26] Beltrami E. (1987): *Mathematics for Dynamic Modeling*, Academic Press, Orlando

[27] Berzins M., Dew P.M. (1991): *Algorithm 690: Chebyshev polynomial software for elliptic-parabolic systems of PDEs*, ACM Transacions on Mathematical Software, Vol. 17, No. 2, 178-206

[28] Bethe H.A., Salpeter E.E. (1977): *Quantum Mechanics of One- and Two-Electron Atoms*, Plenum Press, New York

[29] Bettenhausen D. (1996): *Automatische Struktursuche für Regler und Strecke*, Fortschrittberichte VDI, Reihe 8, Nr. 474, VDI, Düsseldorf

[30] Betts J.T. (1997): *Experience with a sparse nonlinear programming algorithm*, in: Large Scale Optimization with Applications, Part II: Optimal Design and Control, L.T. Biegler, T.F. Coleman, A.R. Conn, F.N. Santos eds., Springer, Berlin

[31] Biegler L.T., Damiano J.J., Blau G.E. (1986): *Nonlinear parameter estimation: a case study comparison*, AIChE Journal, Vol. 32, No. 1, 29-45

[32] Bird H.A., Milliken G.A. (1976): *Estimable functions in the nonlinear model*, Communications of Statistical Theory and Methods, Vol. 15, 513-540

[33] Bird, R.B., Stewart W.E., Lightfoot E.N. (1960): *Transport Phenomena*, John Wiley, New York

[34] Birk J., Liepelt M., Schittkowski K., Vogel F. (1999): *Computation of optimal feed rates and operation intervals for turbular reactors*, Journal of Process Control, Vol. 9, 325-336

[35] Bitterlich S., Knabner P. (2003): *Experimental design for outflow experiments based on a multi-level identification method for material laws*, Inverse Problems, Vol. 19, 1011-1030

[36] Black F., Scholes M. (1973): *The pricing of options and corporate liabilities*, Journal of Political Economics, Vol. 81, 637-659

[37] Blom J.G., Zegeling P.A. (1994): *Algorithm 731: A moving grid interface for systems of one-dimensional time-dependent partial differential equations*, ACM Transactions on Mathematical Software, Vol. 20, No. 2, 194-214

[38] Bock H.G. (1978): *Numerical solution of nonlinear multipoint boundary value problems with applications to optimal control*, Zeitschrift für Angewandte Mathematik und Mechanik, Vol. 58, 407

[39] Bock H.G. (1983): *Recent advantages in parameter identification techniques for ODE,* Proceedings of the International Workshop on *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, Birkhäuser, Boston, Basel, Berlin 95-121

[40] Boderke P., Schittkowski K., Wolf M., Merkle H.P. (2000): *A mathematical model for diffusion and concurrent metabolism in metabolically active tissue*, Journal on Theoretical Biology, Vol. 204, No. 3, 393-407

[41] Bojkov B., Hansel R., Luus R. (1993): *Application of direct search optimization to optimal control problems*, Hungarian Journal of Industrial Chemistry, Vol. 21, 177-185

[42] Borggaard J., Burns J. (1997): *A PDE sensitivity method for optimal aerodynamic design*, Journal of Computational Physics, Vol. 136, No. 2, 366-384

[43] Bossel H. (1992): *Modellbildung und Simulation*, Vieweg, Braunschweig

[44] Box G.P., Hunter W.G., MacGregor J.F., Erjavec J. (1973): *Some problems associated with the analysis of multiresponse data*, Technometrics, Vol. 15, 33-51

[45] Box G.P., Hunter W.G., Hunter J.S. (1978): *Statistics for Experimenters*, John Wiley, New York

[46] Bryson A.E., Denham W.F., Dreyfus S.E. (1963): *Optimal programming problems with inequality constraints*, AIAA Journal, Vol. 1, No. 11, 2544-2550

[47] Bryson A.E., Ho Y.C. (1975): *Applied Optimal Control*, Hemisphere, New York

[48] Buchauer O., Hiltmann P., Kiehl M. (1992): *Sensitivity analysis of initial-value problems with applications to shooting techniques,* DFG-SPP-Report No. 403, Mathematisches Institut, TU München

[49] Buwalda J.G., Ross G.J.S., Stribley D.B., Tinker P.B. (1982): *The development of endomycorrhizal root systems*, New Phytologist, Vol. 92, 391-399

[50] Bykov V., Yablonskii G., Kim V. (1978): *On the simple model of kinetic self-oscillations in catalitic reaction of CO oxidation*, Doklady AN USSR, Vol. 242, 637-639

[51] Caassen N., Barber S.A. (1976): *Simulation model for nutrient uptake from soil by a growing plant root system*, Agronomy Journal, Vol. 68, 961-964

[52] Campbel J.H. (1976): *Pyrolysis of subbituminous coal as it relates to in situ gasification, Part 1: Gas evalution*, Report UCRL-52035, Lawrence Livermore Lab., Livermore, USA

[53] Caracotsios M., Stewart W.E. (1985): *Sensitivity analysis of initial value problems with mixed ODE's and algebraic equations*, Computers and Chemical Engineering, Vol. 9, 359-365

[54] Caracotsios M., Stewart W.E. (1995): *Sensitivity analysis of initial-boundary-value problems with mixed PDE's and algebraic equations*, Computers and Chemical Engineering, Vol. 19, 1019-1030

[55] Carrasco E.F., Banga J.R. (1998): *A hybrid method for the optimal control of chemical processes*, Report, Chemical Engineering Lab., CSIC, University of Vigo, Spain

[56] Cash J.R., Karp A.H. (1990): *A variable order Runge-Kutta method for Initial values: problems with rapidly varying right-hand sides*, ACM Transactions on Mathematical Software, Vol. 16, No. 3, 201-222

[57] Chang K.S. (1978): *Second-order computational methods for distributed parameter optimal control problems*, in: Distributed Parameter Systems, W.H. Ray, D.G. Lainiotis eds., Marcel Dekker, New York, Basel, 47-134

[58] Chemburkar R.M., Morbidelli M., Varma A. (1986): *Parametric sensitivity of a CSTR*, Chemical Engineering Science, Vol. 41, 1647

[59] Chen G., Mills W.H. (1981): *Finite elements and terminal penalization for quadratic cost optimal control problems governed by ordinary differential equations*, SIAM Journal on Control and Optimization, Vol. 19, 744-764

[60] Cherruault Y. (1986): *Explicit and numerical methods for finding optimal therapeutics*, Mathematical Modelling, Vol. 7, 173-183

[61] Chicone C. (1999): *Ordinary Differential Equations with Applications*, Springer, New York

[62] Chudej K., Petzet V., Scherdel S., Pesch H.J., Schittkowski K., Heidebrecht P., Sundmacher K. (2003): *Numerical simulation of a 1D model of a molten carbonate fuel cell*, PAMM-Proceedings of Applied Mathematics and Mechanics, Vol. 3, 521-522

[63] Collatz L. (1960): *The Numerical Treatment of Differential Equations*, Springer, Berlin

[64] Colombeau J.F., Le Roux (1986): *Numerical techniques in elastoplasticity*, in: Nonlinear Hyperbolic Problems, C. Carasso, P.-A. Raviart, D. Serre eds., Lecture Notes in Mathematics, No. 1270, Springer, Berlin

[65] Crank J. (1970): *The Mathematics of Diffusion,* Oxford at the Clarendon Press

[66] Cuthrell J.E., Biegler L.T. (1989): *Simultaneous optimization nad solution methods for batch reactor control profiles*, Computational Chemical Engineering, Vol. 13, 49-62

[67] Dahlquist G., Edsberg L., Sköllermo G., Söderlind G. (1982): *Are the numerical methods and software satisfactory for chemical kinetics?*, in: Numerical Integration of Differential Equations and Large Linear Systems, J. Hinze ed., Springer, Berlin

[68] Daniel C., Wood F.S. (1980): *Fitting Equations to Data*, John Wiley, New York

[69] Davidian M., Giltinan D.M. (1995): *Nonlinear Models for Repeated Measurement Data*, Chapman and Hall, London

[70] Davis H.T. (1962): *Introduction to Nonlinear Differential and Integral Equations*, Dover

[71] Denbigh K.G. (1958): *Optimum temperature sequence in reactors*, Chemical Engineering Sciences, Vol. 8, 125-132

[72] Dennis J.E.jr., Heinkenschloss M., Vicente L.N. (1998): *Trust-region interior-point SQP algorithm for a class of nonlinear programming problems*, SIAM Journal on Control, Vol. 36, No. 5, 1750-1794

[73] Diener I. (1986): *Trajectory nets connecting all critical points of a smooth function*, Mathematical Programming, Vol. 36, 340-352

[74] Dietrich E.E., Eigenberger G. (1996): *Compact finite difference methods for the solution of chemical engineering problems*, in: Scientific Computing in Chemical Engineering, Keil, Mackens, Voss, Werther eds., Springer, Berlin

[75] Dobmann M., Liepelt M., Schittkowski K. (1995): *Algorithm 746: PCOMP: A Fortran code for automatic differentiation,* ACM Transactions on Mathematical Software, Vol. 21, No. 3, 233-266

[76] Dobmann M., Liepelt M., Schittkowski K., Traßl C. (1995): *PCOMP: A Fortran code for automatic differentiation, language description and user's guide,* Report, Dept. of Mathematics, University of Bayreuth, Germany

[77] Dobmann M., Schittkowski K. (1995): *PDEFIT: A Fortran code for constrained parameter estimation in partial differential equations, - user's guide* -, Report, Dept. of Mathematics, University of Bayreuth, Germany

[78] Dolan E.D., Moré J. (2001): *Benchmarking optimization software with COPS*, Technical Report ANL/MCS-246, Argonne National Laboratory, Mathematics and Computer Science Division, Argonne, Illinois

[79] Donea J., Huerta A. (2003): *Finite Element Methods for Flow Problems*, Wiley

[80] Dunn I.J., Heinzle E., Ingham J., Prenosil J.E. (1992): *Biological Reaction Engineering*, VCH, Weinheim

[81] Dwyer H.A., Sanders B.R. (1978): *Numerical modeling of unsteady flame propagation*, Acta Astronautica, Vol. 5, 1171-1184

[82] Edgar T.F., Himmelblau D.M. (1988): *Optimization of Chemical Processes,* McGraw Hill, New York

[83] Edgar T.F., Lapidus L. (1972): *The computation of optimal singular bang-bang control II. Nonlinear systems*, AIChE Journal, Vol. 18, 780-785

[84] Edsberg L., Wedin P.A. (1995): *Numerical tools for parameter estimation in ODE-systems*, Optimization Methods and Software, Vol. 6, 193-218

[85] Ehrig R., Nowak U., Oeverdieck L., Deuflhard P. (1999): *Advanced extrapolation methods for large scale differential algebraic problems*, in: High Performance Scientific and Engineering Computing, H.-J. Bungartz, F. Durst, and Chr. Zenger (eds.), Lecture Notes in Computational Science and Engineering, Springer, Vol. 8, 233-244

[86] Eich-Soellner E., Führer C. (1998): *Numerical Methods in Multibody Dynamics*, Teubner, Stuttgart

[87] Eigenberger G., Butt J.B. (1976): *A modified Crank-Nocolson technique with non-equidistant space steps*, Chemical Engineering Sciences, Vol. 31, 681-691

[88] Ekeland K., Owren B., Oines E. (1998): *Stiffness detection and estimation of dominant spectra with explicit Runge-Kutta methods*, ACM Transaction on Mathematical Software, Vol. 24, No. 4, 368-382

[89] Elezgaray J., Arneodo A. (1992): *Crisis induced intermittent bursting in reaction-diffusion chemical systems*, Physical Reviews Letters, Vol. 68, 714-717

[90] Elnagar G.N., Kazemi M.A. (1998): *Pseudospectral Chebyshev optimal control of constrained nonlinear dynamical systems*, Computational Optimization and Applications, Vol. 11, No. 2, 195-213

[91] Engleborghs K., Lust K., Roose D. (1999): *Direct computation of periodic doubling bifurcation points of large-scale systems of ODE's using a Newton-Picard method*, IMA Journal of Numerical Analysis, Vol. 19, 525-547

[92] Engquist, B. (1986): *Computation of oscillatory solutions to partial differential equations*, in: C. Carasso, P.-A. Raviart, D. Serre eds., Nonlinear Hyperbolic Problems, Lecture Notes in Mathematics, No. 1270, Springer, Berlin

[93] Enright W.H., Hull T.E. (1976): *Comparing numerical methods for the solution of stiff systems of ODEs arising in chemistry*, in: Numerical Methods for Differential Systems, L. Lapidus, W.E. Schiesser eds., Academic Press, New York, 45-66

[94] Farnia K. (1976): *Computer-assisted experimental and analytical study of time/temperature-dependent thermal properties of the aluminium alloy 2024-T351*, Ph.D. Thesis, Dept. of Mechanical Engineering, Michigan State University

[95] Fathi H., Schittkowski K., Hamielec A.E. (2004): *Dynamic modelling of living anionic solution polymerization of styrene/butadiene/divinyl benzene in a continuous stirred tank reactor train*, Polymer-Plastics Technology and Engineering Vol. 43, No. 3, 571-613

[96] Feldman H.A. (1972): *Mathematical theory of complex ligand-binding systems at equilibrium: Some methods for parameter fitting*, Analytical Biochemistry, Vol. 48, 317-338

[97] Fedorov V.V. (1972): *Theory of Optimal Experiments*, Academic Press, New York

[98] Fermi E., Ulam S., Pasta J. (1974): *Studies of nonlinear problems I*, in: Nonlinear Wave Motion, Lectures on Applied Mathematics, AMS, Vol. 15, 143-155

[99] Findeisen R., Allgöwer F. (2000): *A nonlinear model predictive control scheme for the stabilization of setpoint families*, Journal A, Benelux Quarterly Journal on Automatic Control, Vol. 41, 187-192

[100] Fischer H. (1991): *Special problems in automatic differentiation,* in: Automatic Differentiation of Algorithms: Theory, Implementation and Application, A. Griewank, G. Corliss eds., SIAM, Philadelphia

[101] Fischer P. (1996): *Modellierung und Simulation der Ammonium- und Nitrat-Dynamik in strukturierten Waldböden under besonderer Berücksichtigung eines dynamischen, hierarchischen Wurzelsystems,* Diploma Thesis, Dept. of Mathematics, University of Bayreuth, Germany

[102] Flaherty J.E., Moore P.K. (1995): *Integrated space-time adaptive hp-refinement methods for parabolic methods*, Applied Numerical Mathematics, Vol. 16, 317-341

[103] Fogler H.S. (1974): *Elements of Chemical Kinetics and Reactor Calculations*, Prentice Hall, Englewood Cliffs, NJ

[104] Frias J.M., Oliveira J.C, Schittkowski K. (2001): *Modelling of maltodextrin DE12 drying process in a convection oven*, Applied Mathematical Modelling, Vol. 24, 449-462

[105] Friedman A., McLead B. (1986): *Blow-up of solutions of nonlinear degenerate parabolic equations*, Archive for Rational Mechanics and Analysis, Vol. 96, 55-80

[106] Fu P.-C, Barford J.P. (1993): *Non-singular optimal control for fed-batch fermentation processes with a differential-algebraic system model*, Journal on Process Control, Vol. 3, No. 4, 211-218

[107] Fujita H. (1975): *Foundations of Ultracentrifugical Analysis*, John Wiley, New York

[108] Galer A.M., Crout N.M.J., Beresford N.A., Howard B.J., Mayes R.W., Barnett C.L., Eayres H., Lamb C.S. (1993): *Dynamic radiocaesium distribution in sheep: measurement and modelling*, Journal of Environmental Radiology, Vol. 20, 35-48

[109] Ganzha V.G., Vorozhtsov E.V. (1996): *Numerical Solutions for Partial Differential Equations*, CRC Press, Boca Raton, New York, London, Tokyo

[110] Geisler J.(1999): *Dynamische Gebietszerlegung für Optimalsteuerungsprobleme auf vernetzten Gebieten unter Verwendung von Mehrgitterverfahren*, Diploma Thesis, Dept. of Mathematics, University of Bayreuth, Germany

[111] Gelmi C., Perez-Correa R., Agosin E. (2003): *Modelling gibberella fujikuroi growth and GA3 production in solid state fermentation*, Report, Department of Chemical and Bioprocess Engineering, Pontificia Universidad Catolica de Chile, Casilla 306, Santiago 22, Chile

[112] Gill P.E., Petzold L., Rosen J.B., Jay L.O., Park K. (1997): *Numerical optimal control of parabolic PDEs using DASOPT*, in: L. T. Biegler, T. F. Coleman, A. R. Conn and F. N. Santosa (eds.), Large Scale Optimization with Applications, Part II: Optimal Design and Control, IMA Volumes in Mathematics and its Applications, Volume 93, 271-300, Springer Verlag, Berlin, Heidelberg and New York

[113] Gonzales-Concepcion C., Pestano-Gabino C. (1999): *Approximated solutions in rational form for systems of differential equations*, Numerical Algorithms, Vol. 21, 185-203

[114] Goodman M.R. (1974): *Study Notes in System Dynamics*, Wright-Allen Press, Cambridge MA.

[115] Goodson R.E., Polis M.P. (1978): *Identification of parameters in distributed systems*, in: Distributed Parameter Systems, W.H. Ray, D.G. Lainiotis eds., Marcel Dekker, New York, Basel, 47-134

[116] Gorfine M., Freedman L., Shahaf G., Mehr R. (2003): *Maximum likelihood ratio test in complex models: An application to B lymphocyte development*, Bulletin of Mathematical Biology, Vol. 65, 1131-1139

[117] Gottwald B.A., Wanner G. (1981): *A reliable Rosenbrock integrator for stiff differential equations*, Computing, Vol. 26, No.2, 355-360

[118] Granvilliers L., Cruz J., Barahona P. (2000): *Parameter estimation using interval computations*, Report, Laboratoire d'Informatique, Universite de Nantes, B.P. 92208, F-44322 Nantes Cedex 3, France

[119] Gray P, Scott S.K. (1990): *Chemical Oscillations and Instabilities*, Clarenden Press

[120] Griewank A., Juedes D., Srinivasan J. (1991): *ADOL-C: A package for the automatic differentiation of algorithms written in C/C++,* Preprint MCS-P180-1190, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, USA

[121] Guckenheimer J., Holmes P. (1986): *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer, New York

[122] Gugat M., Leugering G., Schittkowski K., Schmidt E.J.P.G. (2001): *Modelling, stabilization and control of flow in networks of open channels*, in: Online Optimization of Large Scale Systems, M. Grötschel, S.O. Krumke, J. Rambau eds., Springer, Berlin, 251-270

[123] Gupta Y.P. (1995): *Bracketing method for on-line solution for low-dimensional nonlinear algebraic equations*, Industrial Engineering and Chemical Research, Vol. 34, 536-544

[124] Haase G. (1990): *Dynamische Simulation einer Destillationskolonne und Entwurf einer Regelung,* Diploma Thesis, Berufsakademie Mannheim

[125] Hadgraft J. (1979): *The epidermal reservoir, a theoretical approach*, International Journal of Pharmaceutics, Vol. 2, 265-274

[126] Hairer E., Nørsett S.P., Wanner G. (1993): *Solving Ordinary Differential Equations I: Nonstiff Problems,* Springer Series Computational Mathematics, Vol. 8, Springer, Berlin

[127] Hairer E., Stoffer D. (1997): *Rerversible long term integration with variable step sizes*, SIAM Journal on Scientific Computing, Vol. 10, 257-269

[128] Hairer E., Wanner G. (1991): *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems,* Springer Series Computational Mathematics, Vol. 14, Springer, Berlin

[129] Hald J., Madsen K. (1981): *Combined LP and quasi-Newton methods for minmax optimization*, Mathematical Programming, Vol. 20, 49-62

[130] Hamdi S., Gottlieb J.J., Hanson J.S. (2001): *Numerical solutions of the equal width wave equation using an adaptive method of lines*, in: Adaptive Methods of Lines, A. Vande Wouwer, Ph. Saucec Ph., W. Schiesser eds., Chapman and Hall/CRC, Boca Raton

[131] Hao D.N., Reinhardt H.-J. (1998): *Gradient methods for inverse heat conduction problems*, in: Inverse Problems in Engineering, Vol. 6, No. 3, 177-211

[132] Hayes B.T., Lefloch P.G. (1998): *Nonclassical shocks and kinetic relations: finite difference schemes*, SIAM Journal on Numerical Analysis, Vol. 35, No. 6, 2169-2194

[133] Heidebrecht P., Sundmacher K. (2003): *Molten carbonate fuel cell (MCFC) with internal reforming: Model-based analysis of cell dynamics*, Chemical Engineering Sciences, Vol. 58, 1029-1036

[134] Heim A. (1998): *Modellierung, Simulation und optimale Bahnplanung von Industrierobotern*, Dissertation, Dept. of Mathematics, Technical University of Munich

[135] Heinzel G., Woloszczak R., Thomann P. (1993): *TOPFIT 2.0: Pharmacokinetic and Pharmacodynamic Data Analysis System,* G. Fischer, Stuttgart, Jena, New York

[136] Hilf K.-D. (1996): *Optimale Versuchsplanung zur dynamischen Roboterkalibrierung*, Fortschritt-Berichte VDI, Reihe 8, Nr. 590

[137] Hines A.L., Maddox R.N. (1985): *Mass Transfer*, Prentice-Hall, Englewood-Cliffs

[138] Hock W., Schittkowski K. (1981): *Test Examples for Nonlinear Programming Codes,* Lecture Notes in Economics and Mathematical Systems, Vol. 187, Springer, Berlin

[139] Hohmann A. (1994): *Multilevel Newton h-p collocation*, ZIB Berlin, Preprint SC 94-25

[140] Hooker P.F. (1965): *Benjamin Gompertz*, Journal of the Institute of Actuaries, Vol. 91, 203-212

[141] Horbelt W., Timmer J., Melzer W. (1998): *Estimating parameters in nonlinear differential equations with application to physiological data*, Report, FDM, University of Freiburg

[142] Houghton D.D., Kasahara A. (1968): *Nonlinear shallow flow over an isolated ridge*, Communications of Pure and Applied Mathematics, Vol. 21, 1-23

[143] Hughes W.F., Brighton J.A. (1991): *Theory and Problems of Fluid Dynamics*, McGraw Hill, New York

[144] Hull T.E., Enright W.H., Fellen B.M., Sedgwick A.E. (1972): *Comparing numerical methods for ordinary differential equations*, SIAM Journal on Numerical Analysis, Vol. 9, 603-637

[145] Igler B., Knabner P. (1997): *Structural identification of nonlinear coefficient functions in transport processes through porous media*, Preprint No. 221, Dept. of Applied Mathematics, University of Erlangen, 1997

[146] Igler B., Totsche K.U., Knabner P. (1997): *Unbiased identification of nonlinear sorption characteristics by soil column breakthrough experiments*, Preprint no. 224, Dept. of Applied Mathematics, University of Erlangen

[147] Ingham J., Dunn I.J., Heinzle E., Prenosil J.E. (1994): *Chemical Engineering Dynamics*, VCH, Weinheim

[148] Jacobson D.H., Mayne D.Q. (1970): *Differential Dynamic Programming*, American Elsevier, New York

[149] Jiang G.-S., Levy D., Lin C.-T., Osher S., Tadmor E. (1997): *High-resolution non-oscillatory schemes with non-staggared grids for hyperbolic conservation laws*, UCLA CAM Report 97-7, Dept. of Mathematics, University of California at Los-Angeles

[150] Jiang G.-S., Shu C.-W. (1995): *Efficient implementation of weighted ENO-methods*, UCLA CAM Report 95-42, Dept. of Mathematics, University of California at Los-Angeles

[151] Johnson C. (1998): *Adaptive finite element methods for conservation laws*, in: Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor eds., Lecture Notes in Mathematics, Vol. 1697, Springer, Berlin

[152] Johnson R.S. (1970): *A nonlinear equation incorporating damping and dispersion*, Journal of Fluid Dynamics, Vol. 42, 49-60

[153] Kahaner D., Moler C., Nash S. (1989): *Numerical Methods and Software*, Prentice Hall, Englewood Cliffs

[154] Kalaba R., Spingarn K. (1982): *Control, Identification, and Input Optimization*, Plenum Press, New York, London

[155] Kaps P., Rentrop P. (1979): *Generalized Runge-Kutta methods of order four with stepsize control for stiff ordinary differential equations,* Numerische Mathematik, Vol. 33, 55-68

[156] Karlsen K.H., Lie K.-A. (1999): *An unconditionally stable splitting scheme for a class of nonlinear parabolic equations*, IMA Journal of Numerical Analysis, Vol. 19, 609-635

[157] Kaps P., Poon S.W.H., Bui T.D. (1985): *Rosenbrock methods for stiff ODE's: A comparison of Richardson extrapolation and embedding techniques*, Computing, Vol. 34, No. 1, 17-40

[158] Kelley C.T. (1999): *Iterative Methods for Optimization*, SIAM, Philadelphia

[159] Kim I., Liebman M.J., Edgar T.F. (1990): *Robust error-in-variables estimation using nonlinear programming techniques*, AIChE Journal, Vol. 36, 985-996

[160] Kletschkowski T., Schomburg U., Bertram A. (2001): *Viskoplastische Materialmodellierung am Beispiel des Dichtungswerkstoffs Polytetrafluorethylen*, Technische Mechanik, Vol. 3, 227-241

[161] Knabner P., van Duijn C.J., Hengst S. (1995): *Crystal dissolution fronts in flows through porous media*, Report, Institute of Applied Mathematics, University of Erlangen

[162] Ko D.Y.C., Stevens W.F. (1971): *Study of singular solutions in dynamic optimization*, AIChE Journal, Vol. 17, 160-166

[163] Kojouharov, Chen B.M. (1999): *Nonstandard methods for the convective-dispersive transport equation with nonlinear reactions*, Numerical Methods for Partial Differential Equations, Vol. 16, No. 1, 107-132

[164] Kowalik J. (1967): *A note on nonlinear regression analysis*, Australian Computational Journal, Vol. 1, 51-53

[165] Kripfganz J., Perlt H. (1994): *Arbeiten mit Mathematica*, Carl Hanser, Oldenburg

[166] Kuzmic P. (1998): *Fixed-point methods for computing the equilibrium composition of complex biochemical mixtures*, Biochemical Journal, Vol. 331, 571-575

[167] Kuzmic P. (1999): *General numerical treatment of competitive binding kinetics: Application to thrombin-dehydrothrombin-hirudin*, Analytical Biochemistry, Vol. 267, 17-23

[168] Kuznetsov, V.A., Puri R.K. (1999): *Kinetic analysis of high-affinity forms of interleukin-13 receptors*, Biophysical Journal, Vol. 77, 154-172

[169] Lafon F., Osher S. (1991): *High order filtering methods for approximating hyperbolic systems of conservation laws*, Journal on Computational Physics, Vol. 96, 110-142

[170] Lambert J.D. (1991): *Numerical Methods for Ordinary Differential Systems: The Initial-Value Problem*, John Wiley, New York

[171] Lanczos C. (1956): *Applied Analysis*, Prentice Hall, Englewood Cliffs

[172] Lagugne-Labarthet F., Bruneel J.L., Sourisseau C., Huber M.R., Börger V., Menzel H. (2002): *A microspectrometric study of the azobenzene chromophore orientation in a holographic diffraction grating inscribed on a p(HEMA-co-MMA) functionalized copolymer film*, Journal of Raman Spectroscopy, Vol. 32, 665-675

[173] Lang J. (1993): *KARDOS: Kascade reaction diffusion one-dimensional system*, Technical Report TR 93-9, ZIB Berlin

[174] Langtangen H.P. (1999): *Computational Partial Differential Equations*, Lecture Notes in Computational Science and Engineering, Vol. 2, Springer, Berlin, Heidelberg

[175] Lapidus, L., Luus, R. (1967): *Optimal Control of Engineering Processes*, Blaisdell, Waltham, Mass.

[176] Lapidus L., Aiken R.C., Liu Y.A. (1973): *The occurence and numerical solution of physical and chemical systems having widely varying time constants*, in: Stiff Differential Systems, E.A. Willoughby ed., Plenum Press, New York, 187-200

[177] Lastman G.J., Wentzell R.A., Hindmarsh A.C. (1978): *Numerical solution of a bubble cavitation problem*, Journal of Computational Physics, Vol. 28, 56-64

[178] Lecar M. (1968): *Comparison of eleven numerical integrations of the same gravitational 25-body problem*, Bulletin Astronomique, Vol. 3, 91

[179] Lee J., Ramirez W.F. (1994): *Optimal fed-batch control of induced foreign protein production by recombinant bacteria*, AIChE Journal, Vol. 40, 899-907

[180] Lee T.T., Wang F.Y., Newell R.B. (1999): *Dynamic modelling and simulation of a complex biological process based on distributed parameter approach*, AIChE Journal, Vol. 45, No. 10, 2245-2268

[181] Lefever R., Nicolis G. (1971): *Chemical instabilities and sustained oscillations*, Journal of Theoretical Biology, Vol. 30, 267-284

[182] Lewis R.M., Patera A.T., Peraire J. (2000): *A posteriori finite element bounds for sensitivity derivatives of partial-differential-equation outputs*, Finite Elements in Design, Vol. 34, 271-290

[183] Lewis J.W. (1994): *Modeling Engineeing Systems*, LLH Technology Publishing, 1994

[184] Liepelt M., Schittkowski K. (2000): *Algorithm 746: New features of PCOMP, a FORTRAN code for automatic differentiation*, ACM Transactions on Mathematical Software, Vol. 26, No. 3, 352-362

[185] Lindberg P.O., Wolf A. (1998): *Optimization of the short term operation of a cascade of hydro power stations*, in: Optimal Contol: Theory, Algorithms, and Applications, W.W. Hager, P.M. Padalos eds., Kluwer Academic Publishers, Dordrecht, Boston, London, 326-345

[186] Lindström P. (1983): *A general purpose algorithm for nonlinear least squares problems with nonlinear constraints,* Report UMINF-103.83, Institute of Information Processing, University of Umea, Umea, Sweden

[187] Lioen W.M., de Swart J.J.B. (1999): *Test set for initial value solvers, Release 2.1*, CWI, Amsterdam, The Netherlands

[188] Liska R., Wendroff B. (1998): *Composite schemes for conservation laws*, SIAM Journal on Numerical Analysis, Vol. 35, No. 6, 2250-2271

[189] Liu X.-D., Osher S. (1997): *Convex ENO high order multi-dimensional schemes without field by field decomposition or staggered grids*, UCLA CAM Report 97-26, Dept. of Mathematics, University of California at Los Angeles

[190] Logan J.D. (1994): *An Introduction to Nonlinear Partial Differential Equations*, John Wiley, New York

[191] Logan J.M. (2001): *Transport Modeling in Hydrochemical Systems*, Interdisciplinary Applied Mathematics, Springer, New York

[192] Lohmann T. (1988): *Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, Dissertation, Dept. of Mathematics, University of Bonn

[193] Lohmann T.W. (1997): *Modellierung und Identifizierung der Reaktionskinetik der Kohlepyrolyse*, Fortschrittsberichte VDI, Reihe 3, No. 499, VDI, Düsseldorf

[194] Lorenz, E.N. (1963): *Deterministic nonperiodic flow*, Journal of Atmospheric Sciences, Vol. 20, 130-141

[195] Loth H., Schreiner T., Wolf M., Schittkowski K., Schäfer U. (2001): *Fitting drug dissolution measurements of immediate release solid dosage forms by numerical solution of differential equations*, Report, Department of Mathematics, University of Bayreuth

[196] Lubich C. (1993): *Integration of stiff mechanical systems by Runge-Kutta methods*, ZAMP, Vol. 44, 1022-1053

[197] Lucht W., Debrabant K. (1996): *Models of quasi-linear PDAEs with convection*, Report, Dept. of Mathematics and Computer Science, University of Halle, Germany

[198] Luenberger D.G. (1979): *Introduction to Dynamic Systems - Theory, Models, and Applications*, John Wiley, New York

[199] Luksan L. (1985): *An implementation of recursive quadratic programming variable metric methods for linearly constrained nonlinear minmax approximations*, Kybernetika, Vol 21, 22-40

[200] Luus R. (1974): *Two-pass method for handling diffcult equality constraints in optimization*, AIChE Journal, Vol. 20, 608-610

[201] Luus, R. (1993): *Optimization of fed-batch fermentors by iterative dynamic programming*, Biotechnology and Bioengineering, Vol. 41, 599-602

[202] Luus R. (1993): *Optimal control of batch reactors by iterative dynamic programming*, Journal of Process Control, Vol. 4, No. 4, 218-226

[203] Luus R. (1998): *Iterative dynamic programming: From curiosity to a practical optimization procedure*, Control and Intelligent Systems, Vol. 26, No. 1, 1-8

[204] Luus R. (2000): *Iterative Dynamic Programming*, Chapman and Hall/CRC, Boca Raton, London, New York, Washington

[205] Luus R. (2002): *Global optimization of Yeos optimal control problem*, Proceedings of the IASTED Conference, Cancun, Mexico, 71-74

[206] Madsen N.K., Sincovec R.F. (1976): *Software for partial differential equations*, in: Numerical Methos for Differential Systems, L. Lapidus, W.E. Schiesser eds., Acedemic Press, New York

[207] Majer C. (1998): *Parameterschätzung, Versuchsplanung und Trajektorienoptimierung für verfahrenstechnische Prozesse*, Fortschrittberichte VDI, Reihe 3, Nr. 538, VDI, Düsseldorf

[208] Majer C., Marquardt W., Gilles E.D. (1995): *Reinitialization of DAE's after discontinouities*, Proceedings of the Fifth European Symposium on Conputer-Aided Process Engineering, 507-512

[209] Maria G. (1989): *An adaptive strategy for solving kinetic model concomitant estimation-reduction problems*, Canadian Journal of Chemical Engineering, Vol. 67, 825-837

[210] Marion M., Mollard A. (1999): *A multilevel characteristics method for periodic convection-dominated diffusion problems*, Numerical Methods for Partial Differential Equations, Vol. 16, No. 1, 107-132

[211] Marquina A, Osher S. (2000): *Explicit algorithms for a new time-dependent model based on level set motion for nonlinear deblurring and noise removal*, Report, Dept. of Mathematics, University of California, Los Angeles

[212] Mattie H., Zhang L.-C., van Strijen E., Razab Sekh B., Douwes-Idema A.E.A. (1997): *Pharmacokinetic and pharmacodynamic models of the antistaphylococcal effects of Meropenem and Cloxacillin in vitro and in experimenatal infection*, Antimicrobial Agents and Chemotherapy, Vol. 41, No. 10, 2083-2088

[213] Maurer H., Weigand M. (1992): *Numerical solution of a drug displacement problem with bounded state variables*, Optimal Control Applications and Methods, Vol. 13, 43-55

[214] Mayer U. (1993): *Untersuchungen zur Anwendung eines Einschritt-Polynom-Verfahrens zur Integration von Differentialgleichungen und DA-Systemen*, Dissertation, Dept. of Chemical Engineering, University of Stuttgart

[215] Mayr L.M., Odefey C., Schutkowski M., Schmid F.X. (1996): *Kinetic analysis of the unfolding and refolding of ribonuclease T1 by a stopped-flow double-mixing technique*, iochemistry, Vol. 35, 5550-5561

[216] Meadows D.H., Meadows D.L., Randers J. (1992): *Beyond the Limits*, Chelsea Green, Post Mills

[217] Meissner E. (2000): *Messung von kurzen Konzentrationsprofilen mit Hilfe der analytischen TEM-EDX am Beispiel der Bestimmung von Diffusionskoeffizienten für Mg-Fe Interdiffusion in Olivin, Dissertation*, Faculty of Biology, Chemistry, and Geological Sciences, University of Bayreuth

[218] Miele A., Wang T., Melvin W.W. (1987): *Optimal abort landing trajectories in the presence of windshear*, Journal of Optimization Theory and Applications, Vol. 12, 815-821

[219] Missel P.J. (2000): *Finite element modeling of diffusion and partioning in biological systems*, Report, Drug Delivery, Alcon Research Ltd., Fort Worth, USA

[220] Mittelmann H.D. (2001): *Sufficient optimality for discretized parabolic and elliptic control problems*, in: Fast Solution of Discretized Optimization Problems, K.-H. Hoffmann, R.H.W. Hoppe, and V. Schulz (eds.), ISNM 138, Birkhäuser, Basel

[221] Mohler R.R., Farooci Z., Heilig T. (1984): *An immune lymphocyte circulation system*, in: System Modelling and Optimization, P. Thoft-Christensen ed., Lecture Notes in Control and Information Sciences, Vol. 59, Springer, 694-702

[222] Molander M. (1990): *Computer aided modelling of distributed parameter process*, Technical Report No. 193, School of Electrical and Computer Engineering, Chalmers University of Technology, Göteborg, Sweden

[223] Moré J.J., Garbow B.S., Hillstrom K.E. (1981): *Testing unconstrained optimization software*, ACM Transactions on Mathematical Software, Vol. 7, No. 1, 17-41

[224] Morrison R.T., Boyd R.N. (1983): *Organic Chemistry*, Allyn and Bacon

[225] Munack A. (1995): *Simulation bioverfahrenstechnischer Prozesse*, in: Prozessimulation, H. Schuler ed., VCH, Weinheim, 409-455

[226] Munack A., Posten C. (1989): *Design of optimal dynamical experiments for parameter estimation*, Proceedings of the Americal Control Conference, Vol. 4, 2010-2016

[227] Müller T.G., Noykova N., Gyllenberg M., Timmer J. (2002): *Parameter identification in dynamical models of anaerobic wastewater treatment*, Mathematical Biosciences, Vol. 177/178, 147-160

[228] Murray, J.D. (1990): *Mathematical Biology*, Springer, New York

[229] Naguma J., Arimoto S., Yoshizawa (1962): *An active pulse transmission line simulating nerve axon*, Proceedings of the IRE, Vol. 50, 2061-2070

[230] Nagurka M.L. (1990): *Fourier-based optimal control of nonlinear dynamic systems*, Journal on Dynamical Systems, Measurements and Control, Vol. 112, 17-26

[231] Nayfeh A. (1972): *Perturbation Analysis*, John Wiley, New York

[232] Neittaanmäki P., Tiba D. (1994): *Optimal Control of Nonlinear Parabolic Systems*, Marcel Dekker, New York, Basel

[233] Nelson W. (1981): *Analysis of performance-degradation data*, IEEE Transactions on Reliability, Vol. 2, No. 2, 149-155

[234] Newell R.B., Lee P.L. (1989): *Applied Process Control - A Case Study*, Prentice Hall, Englewood Cliffs, New Jersey

[235] Nowak U. (1995): *A fully adaptive MOL-treatment of parabolic 1D-problems with extrapolation techniques*, Preprint SC 95-25, ZIB Berlin

[236] Noykova N., Müller T.G., Gyllenberg M., Timmer J. (2001): *Quantitative analysis of anaerobic wastewater treatment processes: Identifiability and parameter estimation*, Biotechnology and Bioengineeriung, Vol. 78, 89-103

[237] Oberle H.J. (1987): *Numerical Computation of Singular Control Functions for a Two-Link Robot Arm*, Lecture Notes in Control and Information Sciences, Vol. 95, Springer, Berlin

[238] Odefey C., Mayr L.M., Schmid F.X. (1995): *Non-prolyl cis-trans peptide bond isomerization as a rate-determining step in protein unfolding and refolding*, Journal of Molecular Biology, Vol. 245, 69-78

[239] Oh S.H., Luus R. (1975): *Optimal feedback control of time-delay systems*, AIChE Journal, Vol. 22, 144-147

[240] Olansky A.S., Deming S.N. (1976): *Optimization and interpretation of absorbance response in the determination of formaldehyde with chromotropic acid*, Analytica Chimical Acta, Vol. 83, 241-249

[241] Osborne M.R. (1972): *Some aspects of nonlinear least squares calculations*, in: Numerical Methods for Nonlinear Optimization, F. Lootsma ed., Academic Press, New York

[242] Otey G.R., Dwyer H.A. (1979): *Numerical study of the interaction of fast chemistry and diffusion*, AIAA Journal, Vol. 17, 606-613

[243] Otter M., Türk S. (1988): *The DFVLR models 1 and 2 of the Manutec R3 robot,* DFVLR-Mitteilungen 88-3, DFVLR, Oberpfaffenhofen, Germany

[244] Ou L.-T. (1985): *2.4-D degradation and 2.4-D degrading microorganisms in soils*, Soil Sciences, Vol. 137, 100-107

[245] Pantelides C.C., Gritsis D., Morison K.R., Sargent R.W.H. (1988): *The mathematical modeling of transient systems using differential-algebraic equations*, Computers and Chemical Engineering, Vol. 12, 440-454

[246] Papalambros P.Y., Wilde D.J. (1988): *Principles of Optimal Design,* Cambridge University Press

[247] Park S., Ramirez W.F. (1988): *Optimal production of secreted protein in fed-batch reactors*, AIChE Journal, Vol. 34, No. 8, 1550-1558

[248] Pennington S.V., Berzins M. (1994): *New NAG Library software for first-order partial differential equations*, ACM Transactions on Mathematical Software, Vol. 20, No. 1, 63-99

[249] Peters N., Warnatz J. eds. (1982): *Numerical Methods in Laminar Flame Propagation*, Notes on Numerical Fluid Dynamics, Vol. 6, Vieweg, Braunschweig

[250] Pfeiffer B.-M., Marquardt W. (1996): *Symbolic semi-discretization of partial differential equation systems*, Mathematics and Computers in Simulation, Vol. 42, 617-628

[251] Pfleiderer J., Reiter J. (1991): *Biplicit numerical integration of partial differential equations with the transversal method of lines*, Report No. 279, DFG SPP *Anwendungsbezogene Optimierung und Steuerung*, Technical University, Dept. of Mathematics, Munich

[252] Pin-Gao Gu, E.T. Vishniac, J.K. Cannizo (2000): *Thermal equilibrium curves and turbulent mixing in Keplerian accretion disks*, The Astrophysical Journal, Vol. 534, 380-397

[253] Pinter J.D. (1995): *Global Optimization in Action*, Kluwer Academic Publishers, Dordrecht, Boston, London

[254] Plusquellec Y., Courbon F., Nogarede S., Houin G. (1998): *Consequence of equal absorption, distribution and/or elimination rate constants*, Report, UFR de Mathematiques, Universite Paul Sabatier, Toulouse

[255] Poeppe C., Pelliciari C., Bachmann K. (1979): *Computer analysis of Feulgen hydrolysis kinetics*, Histochemistry, Vol. 60, 53-60

[256] Posten C., Munack A. (1989): *On-line application of parameter estimation accuracy to biotechnical processes*, Proceedings of the Americal Control Conference, Vol. 3, 2181-2186

[257] Price H., Varga R., Warren J. (1966): *Application of oscillation matrices to diffusion-convection equations*, Journal of Methematical Physics, 301-311

[258] Räumschüssel S. (1998): *Rechnerunterstützte Vorverarbeitung und Codierung verfahrenstechnischer Modelle für die Simulationsumgebung DIVA*, Fortschrittberichte VDI, Reihe 20, Nr. 270, VDI, Düsseldorf

[259] Radu F.A., Bause M., Knabner P., Lee G.W., Friess W.C. (2001): *Drug release from collagen matrices*, Preprint IAM Erlangen Nr. 284, University of Erlangen, D- 91058 Erlangen

[260] Ratkowsky D.A. (1988): *Nonlinear Regression Modeling*, Marcel Dekker, New York

[261] Reich J.G.,. Zinke I. (1974): *Analysis of kinetic and binding measurements, IV Redundancy of model parameters*, Studia Biophysics, Vol. 43, 91-107

[262] Rektorys K. (1982): *The Method of Discretization in Time and Partial Differential Equations*, Reidel, Dordrecht

[263] Rhee, K.I., Sohn H.Y. (1989): *The selective carbochlorination of iron from titaniferous magnetite ore in a fluidized bed*, Matallurgical Transactions B, Vol. 21B, 341-347

[264] Richter O., Diekkrueger B., Noertersheuser P. (1996): *Environmental Fate Modelling of Pesticides*, VCH, Weinheim

[265] Richter O., Noertersheuser, Pestemer W. (1992): *Non-linear parameter estimation in pesticide degradation*, The Science of the Total Environment, Vol. 123/124, 435-450

[266] Richter O., Söndgerath D. (1990): *Parameter Estimation in Ecology*, VCH, Weinheim

[267] Richter O., Spickermann U., Lenz F. (1991): *A new model for plant growth*, Gartenbauwissenschaft, Vol. 56, No. 3, 99-106

[268] Ricker W. (1954): *Stock and recruitment*, Journal of Fishery Research Board Canada, Vol. 211, 559-663

[269] Robertson H.H. (1966): *The solution of a set of reaction rate equations*, in: Numerical Analysis, J. Walsh ed., Academic Press, London, New York, 178-182

[270] Rominger K.L., Albert H.J. (1985): *Radioimmunological determination of Fenoterol. Part I: Theoretical fundamentals,* Arzneimittel-Forschung/Drug Research, Vol. 35, No. 1, 415-420

[271] Rosenau P., Ströder A.C., Stirbet A.D., Strasser R.J. (1999): *Recent advances in modelling the photosynthesis*, Report, IWR, University of Heidelberg

[272] Ross G.J.S. (1990): *Nonlinear Estimation*, Springer, Berlin

[273] Saad M.F., Anderson R.L., Laws A., Watanabe R.M., Kades W.W., Chen Y.-D.I. , Sands R.E., Pei D., Bergmann R.N. (1994): *A comparison between the minimal model and the glucose clamp in the assessment of insulin sensitivity across the spectrum of glucose tolerance*, Diabetes, Vol. 43, 1114-1121

[274] Sakawa Y., Shindo Y. (1982): *Optimal control of container cranes*, Automatica, Vol. 18, 257-266

[275] Sanz-Serna J.M., Calvo M.P. (1994): *Numerical Hamiltonian Processes*, Chapman and Hall, London

[276] Schenk J.L., Staudinger G. (1989): *Computer model of pyrolysis for large coal particles*, in: Proceedings of the International Conference of Coal Science, Tokyo

[277] Schiesser W.E. (1991): *The Numerical Method of Lines,* Academic Press, New York, London

[278] Schiesser W.E. (1994): *Computational Mathematics in Engineering and Applied Science*, CRC Press, Boca Raton

[279] Schiesser W.E. (1994): *Method of lines solution of the Korteweg-de Vries equation*, Computers in Mathematics and Applications, Vol. 28, No. 10-12, 147-154

[280] Schiesser W.E., Silebi C.A. (1997): *Computational Transport Phenomena*, Cambridge University Press

[281] Schittkowski K. (1979): *Numerical solution of a time-optimal parabolic boundary-value control problem*, Journal of Optimization Theory and Applications, Vol. 27, 271-290

[282] Schittkowski K. (1987): *More Test Examples for Nonlinear Programming,* Lecture Notes in Economics and Mathematical Systems, Vol. 182, Springer, Berlin

[283] Schittkowski K. (1992): *Solving nonlinear programming problems with very many constraints*, Optimization, Vol. 25, 179-196

[284] Schittkowski K. (1997): *Parameter estimation in one-dimensional time dependent partial differential equations*, Optimization Methods and Software, Vol. 7, No. 3-4, 165-210

[285] Schittkowski K. (2002): *Numerical Data Fitting in Dynamical Systems - A Practical Introduction with Applications and Software*, Kluwer Academic Publishers, Dordrecht, Boston, London

[286] Schittkowski K. (2004): *PCOMP: A modeling language for nonlinear programs with automatic differentiation*, in: *Modeling Languages in Mathematical Optimization*, J. Kallrath ed., Kluwer, Norwell, MA, 349-367

[287] Schittkowski K. (2007): *Experimental design tools for ordinary and algebraic differential equations*, Industrial and Engineering Chemistry Research, Vol. 46, 9137-9147

[288] Schittkowski K. (2009): *Data fitting and experimental design in dynamical systems with EASY-FIT ModelDesign - user's guide*, Report, Department of Computer Science, University of Bayreuth

[289] Schittkowski T., Brüggemann, Mewes B. (2002): *LII and Raman measurements in sooting methane and ethylene flames*, Report, LTTT, Dept. of Applied Natural Sciences, University of Bayreuth

[290] Schreiner T. (1995): *Mechanistische und kinetische Parameter der Arzneistoffauflösung aus festen Zubereitungen als Kriterien der galenischen Qualitätssicherung, Dissertation*, Dept. of Pharmaceutics, University of Saarbrücken

[291] Schumacher E. (1997): *Chemische Reaktionskinetik*, Script, Dept. of Chemistry, University of Bern, Switzerland

[292] Schwartz A.L. (1996): *Theory and implementation of numerical methods based on Runge-Kutta integration for solving optimal control problems*, Ph.D. Thesis, Dept. of Electrical Engineering and Computer Sciences, University of California at Berkeley

[293] Scott M.R., Watts H.A. (1976): *Solution methods for stiff differential equations*, in: Numerical Methods for Differential Systems, L. Lapidus, W.E. Schiesser eds., Academic Press, New York, London, 197-227

[294] Seber G.A.F., Wild C.J. (1989): *Nonlinear Regression*, John Wiley, New York

[295] Seelig F.F. (1981): *Unrestricted harmonic balance II. Application to stiff ODE's in enzyme catalysis*, Journal of Mathematical Biology, Vol. 12, 187-198

[296] Seifert P. (1990): *A realization of the method of lines used for chemical problems*, Colloquia Mathematica Societatis Janos Bolyai, Numerical Methods, Vol. 59, 363-373

[297] Sellers P.J., Dickinson R.E., Randall D.A., Betts A.K., Hall F.G., Berry J.A.,Collatz G.J., Denning A.S., Mooney H.A., Nobre C.A., Sato N., Field C.B., Henderson-Sellers A. (1997): *Modeling the exchanges of energy, water, and carbon between continents and the atmosphere*, Science, Vol. 275, 502-509

[298] Severo A.M., Diaz-Romanach M.L.B., Rodriguez L.M.P., Ginart J.B. (2004): *Numerical experience in the solution of several kind estimation problems in dynamical systems*, International Conference on Modeling and Optimization, MODOPT 2003, Tmuco, Chile, 16.-22.1.2004

[299] Seydel R. (1988): *From Equilibrium to Chaos: Practical Bifurcation and Stability Analysis*, Elsevier, Amsterdam

[300] Shacham M. (1985): *Comparing software for the solution of systems of nonlinear algebraic equations arising in chemical engineering*, Computers and Chemical Engineering, Vol. 9, 103-112

[301] Shakhno, S. (2001): *Some numerical methods for nonlinear least squares problems*, in: Symbolic Algebraic Methods and Verification Methods, Alefeld, Götz eds., Springer, Wien, 235-249

[302] Shampine L.F. (1980): *Evaluation of a test set for stiff ODE solvers*, ACM Transactions on Mathematical Software, Vol. 7, No. 4, 409-420

[303] Shampine L.F. (1994): *Numerical Solution of Ordinary Differential Equations*, Chapman and Hall, New York, London

[304] Shampine L.F., Watts H.A., Davenport S.M. (1976): *Solving nonstiff ordinary differential equations - The state of the art*, SIAM Reviews, Vol. 18, 376-411

[305] Shampine L.F., Watts H.A. (1979): *The art of writing a Runge-Kutta code,* Applied Mathematics and Computations, Vol. 5, 93-121

[306] Shampine L.F., Gordon M.K. (1975): *Computer Solution of Ordinary Differential Equations: The Initial-Value Problem*, Freeman, San Francisco

[307] Sheng Q., Khalic A.Q.M. (1999): *A compound adaptive approach to degenerate nonlinear quenching problems*, Numerical Methods for Partial Differential Equations, Vol. 16, No. 1, 107-132

[308] Simeon B., Grupp F., Führer C., Rentrop P. (1994): *A nonlinear truck model and its treatment as a multibody system*, Journal of Computational and Applied Mathematics, Vol. 50, 523-532

[309] Sincovec R.F., Madsen N.K. (1975): *Software for nonlinear partial differential equations*, ACM Transactions on Mathematical Software, Vol. 1, No. 3, 232-260

[310] Slider H.C. (1976): *Practical Petroleum Reservoir Engineering Methods*, The Petroleum Publishing Company, Tulsa, Oklahoma

[311] Spicer P.T., Pratsinis S.E. (1996): *Coagulation and fragmentation: universal steady-state particle-size distribution*, AICHE Journal, Vol. 42, No. 6, 1612-1620

[312] Spoelstra J., van Wyk D.J. (1987): *A method of solution for a non-linear diffusion model and for computing the parameters in a model*, Journal of Computational and Applied Mathematics, Vol. 20, 379-385

[313] Stehfest H. (1970): *Algorithm 368: Numerical inversion of Laplace transforms,* Communications of the ACM, Vol. 13, 47-49

[314] Steinebach G., Rentrop P. (2000): *An adaptive method of lines approach for modelling flow and transport in rivers*, Preprint No. 00/09, IWRMM, University of Karlsruhe

[315] Steinsträsser I. (1994): *The organized HaCaT cell culture sheet: A model approach to study epidermal peptide drug metabolism,* Dissertation, Pharmaceutical Institute, ETH Zürich

[316] Stenger F., Gustafson S.-A., Keyes B., O'Reilly M., Parker K. (1999): *ODE-IVP-PACK via Sinc indefinite integration and Newton's method*, Numerical Algorithms, Vol. 20, 241-268

[317] Stirbet A.D., Strasser R.J. (1996): *Numerical solution of the in vivo fluorescence in plants*, Mathematical Computations and Simulations, Vol. 42, 245-253

[318] Stoer J., Bulirsch R. (1980): *Introduction to Numerical Analysis*, Springer, New York

[319] Stortelder W.J.H. (1998): *Parameter estimation in nonlinear dynamical systems*, Dissertation, National Research Institute for Mathematics and Computer Science, University of Amsterdam

[320] Strikwerda J.C. (1997): *Finite Difference Schemes and Partial Differential Equations*, Chapman and Hall, New York

[321] Swameye I., Müller T.G., Timmer J., Sandra O., Klingmüller U. (2002): *Identification of nucleocytoplasmic cycling as a remote censor in cellular signaling by database modeling*, Proceedings of the National Society of Sciences of the USA, Vol. 100, 1028-1033

[322] Tanaka Y., Fukushima M., Ibaraki T. (1988): *A comparative study of several semi-infinite nonlinear programming algorithms*, European Journal of Operations Research, Vol. 36, 92-100

[323] Teo K.L., Wong K.H. (1992): *Nonlinearly constrained optimal control of nonlinear dynamic systems*, Journal of the Australian Mathematical Society, Ser. B, Vol. 33, 507-530

[324] Thomas J.W. (1995): *Numerical Partial Differential Equations*, Texts in Applied Mathematics, Vol. 22, Springer, Berlin

[325] Thomaseth K., Cobelli C. (1999): *Generalized sensitivity functions in physiological system identification*, Annals of Biomedical Engineering, Vol. 27, 607-616

[326] Thomopoulus S.C.A., Papadakis I.N.M. (1991): *A single shot method for optimal step computation in gradient algorithms*, Proceedings of the 1991 American Control Conference, Boston, MA, American Control Counciol, IEEE Service Center, Piscataway, NJ, 2419-2422

[327] Timmer J., Rust H., Horbelt W., Voss H.U. (2000): *arametric, nonparametric and parametric modelling of a chaotic circuit time series*, Physics Letters A, Vol. 274, 123-134

[328] Timoshenko S., Goodier J.N. (1970): *Theory of Elasticity*, McGraw Hill, New York

[329] Tjoa I., Biegler L. (1991): *Simultaneous solution and optimization strategies for parameter estimation of differantial-algebraic equation systems*, Industrial Engineering Chemistry Research, Vol. 30, 376-385

[330] Tjoa T.B., L.T. Biegler L.T. (1992): *Reduced successive quadratic programming strategy for errors-in-variables estimation*, Computers and Chemical Engineering, Vol. 16, 523

[331] Troeltzsch F. (1999): *Some remarks on second order sufficient optimality conditions for nonlinear elliptic and parabolic control problems*, in: Proceedings of the Workshop 'Stabilität und Sensitivität von Optimierungs- und Steuerungsproblemen', Burg (Spreewald), Germany, 21.-23.4.99

[332] Tveito A., Winther R. (1998): *Introduction to Partial Differential Equations*, Springer, New York

[333] Tzafrini A.R. (2000): *Mathematical modelling of diffusion-mediated release from bulk degrading matrices*, Journal of Controlled Release, Vol. 63, 69-79

[334] Ulbrich S. (1995): *Stabile Randbedingungen und implizite entropiedissipative numerische Verfahren für Anfangs-Randwertprobleme mehrdimensionaler nichtlinearer Systeme von Erhaltungsgleichungen mit Entropie*, Dissertation, TU München, Institut für Angewandte Mathematik und Statistik

[335] van den Bosch B.A.J. (1978): *Identification of parameters in distributed chemical reactors*, in: Distributed Parameter Systems, W.H. Ray, D.G. Lainiotis eds., Marcel Dekker, New York, Basel, 47-134

[336] van Doesburg H., De Jong W.A. (1974): *Dynamic behavior of an adiabatic fixed-bed methanator*, in: Advances in Chemistry, Vol. 133, International Symposium on Reaction Engineering, Evanston, 489-503

[337] van Duijn C.J. (1989): *Flow through porous media*, DFG-SPP-Report No. 135, Dept. of Mathematics, University of Augsburg

[338] van Genuchten M.T. (1980): *A closed-form equation for predicting the hydraulic conductivity of unsaturated soils*, Soil Science Society of America Journal, Vol. 44, 892-898

[339] van Genuchten M.T., Wierenga P.J. (1976): *Mass transfer studies in sorbing porous media. 1. Analytical solutions,* Soil Sciences Society of America Journal, Vol. 44, 892-898

[340] van Kan J.J.I.M., Segal A. (1995): *Numerik partieller Differentialgleichungen für Ingenieure,* Teubner

[341] Vande Wouwer A., Saucec Ph., Schiesser W.E. (2001): *Adaptive Methods of Lines*, Chapman and Hall/CRC, Boca Raton

[342] Varah J.M. (1982): *A spline least squares method for numerical parameter estimation in differential equations*, SIAM Journal on Scientific Statistical Computing, Vol. 3, 28-46

[343] Varma A., Morbidelli M., Wu H. (1999): *Parametric Sensitivity in Chemical Systems*, Cambridge University Press

[344] Vasile M., Jehn R. (1999): *Low thrust orbital transfer of a LISA spacecraft with constraints on the solar aspect angle*, MAS Working Paper 424, ESOC, Darmstadt

[345] Vassiliadis V.S., Sargent R.W.H., Pantelides C.C. (1994): *Solution of a class of multistage dynamic optimization problems, 2. Problems with path constraints*, Industrial Engineering and Chemical Research, Vol. 33, No. 9, 2123-2133

[346] Verwer J.G., Blom J.G., Furzeland R.M., Zegeling P.A. (1989): *A moving grid method for one-dimensional PDEs based on the method of lines*, in: Adaptive Methods for Partial Differential Equations, J.E. Flaherty, P.J. Paslow, M.S. Shephard, J.D. Vasilakis eds., SIAM, Philadelphia, Pa., 160-175

[347] Verwer J.G., Blom J.G., Sanz-Serna J.M. (1989): *An adaptive moving grid method for one-dimensional systems of partial differential equations*, Journal of Computational Physics, Vol. 82, 454-486

[348] Vlassenbeck J., van Dooren R. (1983): *Estimation of the mechanical parameters of the human respiratory system*, Mathematical Biosciences, Vol. 69, 31-55

[349] von Stryck O. (1995): *Numerische Lösung optimaler Steuerungsprobleme*, Fortschrittsberichte VDI, Reihe 8, Nr. 441, VDI, Düsseldorf

[350] Vossen G., Rehbock V., Siburian A. (2007): *Numerical solution methods for singular control with multiple state dependent forms*, Optimization Methods and Software, Vol. 22, 551-559

[351] Vrar, C.K. (2000): *The study of kinetics and mechanism of chlorination of copper(I)sulphide by chlorine in the presence of oxygen*, Canadian Metallurgical Quarterly, Vol. 39, 163-174

[352] Vreugdenhil C.B., Koren B. eds. (1993): *Numerical Methods for Advection-Diffusion Problems*, Vieweg, Braunschweig

[353] Walas S.M. (1991): *Modeling with Differential Equations in Chemical Engineering*, Butterworth-Heinemann, Boston

[354] Walter E., Pronzato L. (1997): *Identification of Parametric Models*, Springer, Paris, Milan, Barcelone

[355] Wang H., Al-Lawatia M., Sharpley R.C. (1999): *A characteristic domain decomposition and space-time local refinement method for first-order linear hyperbolic equations with interface*, Numerical Methods for Partial Differential Equations, Vol. 15, No. 1, 1-28

[356] Watts D.G. (1981): *An introduction to nonlinear least squares*, in: Kinetic Data Analysis: Design and Analysis of Enzyme and Pharmacokinetic Experiments, L. Endrenyi ed., Plenum Press, New York, 1-24

[357] Weinreb A., Bryson A.E.Jr. (1985): *Optimal control of systems with hard control bounds*, IEEE Transactions on Automatic Control, Vol. AC-30, 1135-1138

[358] Weizhong D., Nassar R. (1999): *A finite difference scheme for solving the heat transport equation at the microscale*, Numerical Methods for Partial Differential Equations, Vol. 15, No. 6, 697-708

[359] Wen C.S., Yen T.F. (1977): *Optimization of oil shale pyrolysis*, Chemical Engineering Sciences, Vol. 32, 346-349

[360] Williams J., Kalogiratou Z. (1993): *Least squares and Chebyshev fitting for parameter estimation in ODE's*, Advances in Computational Mathematics, Vol. 1, 357-366

[361] Willoughby, R.A. (1974): *Stiff Differential Systems*, Plenum Press, New York

[362] Wolf M. (1994): *Mathematisch-physikalische Berechnungs- und Simulationsmodelle zur Beschreibung und Entwicklung transdermaler Systeme,* Habilitationsschrift, Mathematisch-Naturwissenschaftliche Fakultät, Universität Bonn

[363] Wolmott P., Dewynne J.N., Howison S.D. (1993): *Option Pricing: Mathematical Models and Computation*, Oxford Financial Press

[364] Wouwer A.V. (1994): *Simulation, parameter and state estimation techniques for distributed parameter systems with real-time application to a multizone furnace*, Dissertation, Faculte Polytechnique de Mons, Belgium

[365] Yang H.Q., Przekwas A.J. (1992): *A comparative study of advanced shock-capturing schemes applied to Burgers' equation*, Journal of Computational Physics, Vol. 102, 139-159

[366] Zeeman E.C. (1972): *Differential equations for the haertbeat and nerve impulse*, in: Towards a Theoretical Biology, C.H. Waddington ed., Edinburgh University Press, Vol. 4, 8-67

[367] Zegeling P.A., Verwer J.G., van Eijkeren J.C.H. (1992): *Application of a moving grid method to a class of 1d brine transport problems in porous media*, International Journal for Numerical Methods in Fluids, Vol. 15, 175-191

[368] Zhengfeng L., Osborne M.R., Prvan T. (2005): *Parameter estimation of ordinary differential equations*, IMA Journal of Numerical Analysis, Vol. 25, 264-285

[369] Zschieschang, T. Dresig, H. (1998): *Zur Zeit-Frequenz-Analyse von Schwingungen in Antrieben von Verarbeitungsmaschinen*, Fortschrittsberichte VDI, Nr. 1416, VDI, Düsseldorf, 489-506

APPENDIX A: Individual Results

This appendix contains a list of all individual results, where we use the abbreviations

name	-	test problem name,
i_f	-	termination reason,
$f(x^\star)$	-	objective function value at final iterate, either sum of squares or result of experimental design run
n_f	-	number of function evaluations,
n_{its}	-	number of gradient evaluations or iterations, respectively,
time	-	calculation time in seconds.

One iteration corresponds to the evaluation of the Jacobian matrix of the data fitting function, and a function call to the evaluation of all data fitting criteria subject to all time and concentration values. The numerical results have been obtained by the codes MODFIT and PDEFIT, see Schittkowski [285], and are listed in Tables 8 and 9.

Table 8: Individual Test Results for MODFIT

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
2BODY	0	0.203262E-02	21	18	0.4844
2CSTR	0	0.357584E-02	19	16	0.2188
2INDVARS	0	0.410029E-05	2	2	0.0000
2LKC_ROB	0	0.320840E-02	86	13	1.9844
2LNK_ROB	0	0.143970E-02	55	45	1.2812
2ND_ORD	0	0.796175E-02	8	8	0.1719
2ND_RATE	0	0.625628E-03	55	55	0.0000
2STGCSTR	0	0.226708E-08	31	30	1.5625
2VALLEYS	0	0.933785E+00	10	9	0.0000
3MODVARS	0	0.603605E+01	56	20	0.0469
4BAR_LNK	0	0.516792E+00	59	19	0.0000
ABSORP	0	0.133967E-01	6	6	0.0625
ACTIVITY	0	0.968376E-01	48	28	0.0625
ACTNITR	0	0.163021E-02	112	60	3.2500
ADI_CSTR	0	0.115602E+00	80	59	0.3125
ADIABATI	0	0.472223E-03	7	7	0.0312
AEKIN	0	0.184005E+00	61	40	0.7188
AEROSOL	0	0.110760E-03	167	36	0.1406
AEROSOLS	0	0.124790E-03	128	55	0.2812
AIRLIFT	0	0.555801E-01	19	19	0.0938
AIRY	0	0.170509E-02	3	3	0.0000
AKTIV_W2	0	0.291967E-02	142	101	1.7344
ALDRIN	0	0.148085E+00	3	3	0.0000
ALPHA_PI	0	0.106304E-02	8	8	0.0312
AMENTO	0	0.513915E+00	155	56	1.0312
AMIDPRO	0	0.654714E-03	6	6	0.2344
AMMONAB	0	0.284643E-02	20	18	0.0156
AMYLASE	0	0.294680E-02	106	69	0.6094
ANAEMEAS	0	0.453652E-02	12	12	0.8594
ANHYD	0	0.318996E-04	21	15	0.1406
ANTIBIO	0	0.123373E-01	158	101	1.7500
APHID	0	0.510077E-01	153	86	2.1562
APHIDEX	0	0.782460E-01	239	94	3.5312
APPRX1	0	0.125341E-05	70	42	0.0312
APPRX2	0	0.200000E+00	9	9	0.0000
APPRX3	0	0.537824E-02	21	21	0.0000
ASPHER_1	0	0.527880E-04	44	11	0.0156
ASPHER_2	0	0.540323E-05	36	24	0.0156
ASPHER_3	0	0.869915E-04	43	28	0.0312

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
ASPHER_4	0	0.136490E-04	7	5	0.0156
ASS_CV1	0	0.230868E+00	61	36	32.6250
ASS_CV2	0	0.116857E-02	20	20	0.0469
ASS_CV3	0	0.751900E-01	22	20	1.0625
ASS_CV4	0	0.635239E-02	26	25	0.1875
ASS_CV5	0	0.431141E-01	82	54	0.3750
ASS_CV6	0	0.551634E-02	209	74	0.4375
ASS_CV7	0	0.941816E-03	132	43	0.5312
ASS_KIN1	0	0.663274E-02	26	20	0.0312
ASS_KIN2	0	0.522608E-02	29	23	0.0156
ASS_KIN3	0	0.208514E-02	14	14	0.0625
ASS_KIN4	0	0.280626E-01	44	34	0.0156
ASS_KIN5	0	0.104381E-01	24	19	0.0938
ASS_KIN6	0	0.522608E-02	29	23	0.0312
ASTRO	0	0.163285E+02	20	12	0.2969
ASYMP	0	0.974844E+00	30	23	0.2188
ATROP_EX	0	0.405124E-02	25	24	0.0000
AXDISP	0	0.354828E-02	24	20	0.2031
B_BLOCK	0	0.859042E-05	5	4	0.0625
B_BLOCK1	0	0.681204E-06	6	4	0.1719
B_BLOCK2	0	0.581542E-18	6	4	1.0938
B_LYMPHO	0	0.114686E+00	97	51	0.3750
BACTERIA	0	0.420754E-02	154	87	3.2344
BALL	0	0.823340E-05	3	3	0.0156
BARN1	0	0.164461E+00	8	8	0.0156
BARN2	0	0.243642E+00	8	8	0.0312
BATCH	0	0.118118E-01	217	100	7.0781
BATCH_CT	0	0.151805E+00	74	73	0.3750
BATCH_E	0	0.752041E-01	85	74	13.0625
BATCH_E1	0	0.416323E-01	80	53	8.5469
BATCH_E2	0	0.427145E-02	98	56	12.0469
BATCH_E3	0	0.207528E-01	90	62	7.0469
BATCH_F1	0	0.117493E-01	31	27	5.4062
BATCH_F2	0	0.561585E-01	27	23	4.7031
BATCH_F3	0	0.700299E-01	77	71	24.9531
BATCH_F4	0	0.134166E+00	39	32	12.7812
BATCH_X1	0	0.426112E-01	5	5	2.8594
BATCH_X2	0	0.243495E+09	21	21	9.3906
BATCHD	0	0.116331E-03	58	36	0.0625
BATCHREA	0	0.703396E-02	4	4	0.2656
BATCOM	0	0.291768E-02	90	53	13.6406
BATEX	0	0.609776E-03	6	6	0.0000
BATFERM	0	0.194244E-02	17	17	0.2500
BATSEG	0	0.106947E-02	11	11	0.0156
BATSEQ	0	0.541191E-02	52	39	0.2344
BCBPLUS	0	0.967194E-02	16	12	16.0781
BEAD	0	0.501979E-02	96	58	0.8594
BEER	0	0.106405E-02	21	21	1.3125
BELLMAN	0	0.162497E-03	9	9	0.0156
BELUSOV	0	0.124201E+01	322	89	13.7500
BENNETT5	0	0.523673E-03	53	14	0.2500
BENZENE	0	0.152954E-01	10	10	0.0000
BENZHYD	0	0.997118E-03	15	11	0.0156
BI_OSC	0	0.145707E-02	4	4	3.7656
BIMOLECU	0	0.666954E-06	63	40	0.0469
BIO_MOD	0	0.169458E-02	13	13	1.7344
BIO_REAC	0	0.241927E+00	358	233	186.0156
BIODEG	0	0.340858E-01	15	15	0.2500

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
BIOKAT	0	0.633111E-03	18	16	0.2969
BIOMASS	0	0.536711E-01	21	17	0.0312
BIOPROC	0	0.890734E-03	73	48	0.5156
BIOREAC	0	0.496625E-04	11	11	0.3906
BIRDMILL	0	0.964619E-03	12	12	0.0000
BITUMEN	0	0.117098E-02	63	42	0.4844
BLASIUS	0	0.160640E-02	10	10	0.3750
BLOOD	0	0.166613E-01	19	12	2.6250
BLOOD_O	0	0.125515E+00	7	6	0.0156
BLOOD_S	0	0.120116E-01	10	10	0.0156
BLYMPH	0	0.528415E-02	84	66	4.5781
BLYMPH_R	0	0.145891E+00	7	7	0.3594
BOGGS2	0	0.325682E-01	40	24	0.0000
BOGGS8	0	0.100000E+01	51	21	0.0000
BOND	0	0.761651E-03	44	37	0.0625
BOXBOD	0	0.116801E+04	10	10	0.0000
BRUNHILD	0	0.708700E-03	13	12	0.1875
BRUSSEL1	0	0.349051E-02	27	23	0.1094
BRUSSEL2	0	0.117238E-02	27	19	0.2188
BSTILL	0	0.141794E-01	17	13	1.2188
BSTILL_I	0	0.128318E-02	8	8	0.8594
BUBBLEC	0	0.553850E-02	26	26	1.3281
BURGER_W	0	0.217141E+00	19	13	0.0000
BVP	0	0.383240E-03	6	6	0.0000
BVP_NL	0	0.405145E-13	6	5	0.0781
BVP4	0	0.442539E-02	27	18	5.0312
CABBAGE	0	0.815741E-02	52	37	0.1875
CAMY	0	0.390568E+00	44	30	1.0469
CAR_CTR1	0	0.887918E+03	48	13	0.3438
CAR_CTR2	0	0.931917E+03	14	12	1.1719
CARGO	0	0.108928E-03	44	44	0.2344
CASC_IMP	0	0.282200E-03	20	11	0.6094
CASCADE1	0	0.175749E-01	56	35	7.1406
CASCADE2	0	0.125829E-01	12	12	0.0312
CASCSEQ	0	0.163235E-02	27	27	0.5000
CASTOR	0	0.142130E-02	14	14	0.2812
CAT_HYD	0	0.183217E-02	4	4	0.0156
CAT_SD	0	0.147937E-03	291	129	19.8125
CAT_SEP	0	0.204039E-02	80	47	0.0000
CATALYST	0	0.952038E-19	18	16	1.0000
CAV_BUBB	0	0.745123E-03	11	11	0.9531
CELLS	0	0.133338E-03	17	15	0.2500
CEMENT	0	0.100648E-03	24	17	0.0469
CENSUS	0	0.202108E+01	49	30	0.0000
CENTRI	0	0.746575E+00	27	27	0.0156
CH_CIRC	0	0.414130E+01	175	51	90.9375
CHAIN_O1	0	0.127133E-02	27	27	0.0312
CHAN_FLO	0	0.207651E-02	36	22	0.1406
CHANNEL	0	0.412075E-03	79	39	0.0781
CHARGE	0	0.354236E-04	35	31	0.4688
CHEM_EQU	0	0.377524E-04	8	8	0.1250
CHEM_OSC	0	0.126732E-01	65	52	0.4688
CHEM_REA	0	0.348959E+00	11	11	0.1250
CHEMO	0	0.163178E-02	16	16	0.8125
CHEMOSTA	0	0.123061E+01	182	99	0.2344
CHEMSTAT	0	0.700717E-03	17	14	0.0312
CHSTA_D	0	0.102231E+00	245	128	2.5938
CHWIRUT1	0	0.238448E+04	17	13	0.5938

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
CHWIRUT2	0	0.513048E+03	17	13	0.0156
CIRCLE	0	0.160976E-02	15	12	4.5625
CIRCUIT	0	0.258727E+02	12	12	26.7969
CLOUD	0	0.669411E-02	32	21	0.2188
CO2_SOL1	0	0.218721E+01	3	3	0.2500
COAL1	0	0.931194E-04	95	69	2.0312
COAL2	0	0.729842E+00	36	31	6.1250
COAL3	0	0.383741E-02	10	10	0.1094
COAL4	0	0.228744E-02	23	18	0.1719
COAL5	0	0.146170E-01	16	14	0.0781
COAL6	0	0.167711E-01	27	25	0.0938
COAL7	0	0.384417E-01	8	8	0.6250
COLCON	0	0.512255E-02	14	14	0.1406
COLLISIO	0	0.167898E+01	115	46	14.0625
COMBPROP	0	0.298156E-13	4	4	0.0000
COMMENSA	0	0.149856E-02	169	104	1.3906
COMP_ADS	0	0.372375E-02	47	42	0.9219
COMP_EXP	0	0.124312E-02	23	23	0.0469
COMPASM	0	0.121793E+00	37	27	1.3750
COMPET	0	0.218667E-01	121	51	0.2500
COMPREAC	0	0.558096E-02	66	37	1.0625
COMPSEG	0	0.575380E-03	16	14	0.2969
CON_BURG	0	0.432463E+00	4	4	0.0000
CONC4	0	0.268968E-02	40	40	0.8750
CONC4A	0	0.157411E-01	38	38	0.7812
CONCS	0	0.256296E-02	8	7	0.0000
CONDENS	0	0.395197E-02	24	23	0.2344
CONF_ALT	0	0.125663E-02	14	14	0.0469
CONFLO1	0	0.154404E-02	6	6	0.0156
CONFLO2	0	0.293630E-02	231	39	1.1719
CONFLO3	0	0.149250E-02	29	18	0.2188
CONINHIB	0	0.596320E-03	129	79	1.8594
CONSREA	0	0.112600E-02	23	14	0.1719
CONSTILL	0	0.341020E-02	82	71	1.6250
CONTCON	0	0.105704E-02	28	22	0.4375
CONTUN	0	0.337635E-04	7	7	0.6406
CONVER	0	0.633840E-10	7	7	0.0625
COOL	0	0.644344E-03	10	8	0.0625
COOL_CRI	0	0.564703E-02	5	5	2.2969
COPPER	0	0.169283E-01	77	12	12.4062
COPPER_D	0	0.980951E+00	39	27	22.8125
COPPER1	0	0.782926E-01	28	27	2.8594
COPPER2	0	0.782927E-01	28	27	2.7812
CR_ELOV	0	0.939013E-03	155	103	3.9062
CRANE	0	0.102126E-04	12	12	0.5469
CS_REAC	0	0.446251E-04	25	25	0.2188
CST_1ORD	0	0.560736E-04	23	20	0.0625
CSTOHNE	0	0.327636E-02	9	8	2.5938
CSTR	0	0.115140E-01	76	41	0.0781
CSTR_BM	0	0.239146E-02	14	14	0.0781
CSTR_CTL	0	0.906784E-04	66	66	4.0938
CSTR_CTR	0	0.599304E-01	40	38	0.5938
CSTR_DFT	0	0.439317E-01	49	34	3.2188
CSTR_JY0	0	0.163618E-01	22	22	6.2188
CSTRCOM	0	0.432515E-02	61	38	0.6719
CUO	0	0.126983E+00	499	145	1.5156
CYC_COM1	0	0.152354E-02	7	7	0.0000
CYC_COM2	0	0.148361E-02	7	7	0.0000

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
DA_X	0	0.290534E+03	14	13	0.7344
DAE_EX	0	0.272178E-03	41	27	2.2344
DAE_I1	0	0.648871E-11	17	11	0.2656
DAE_I2	0	0.191196E-09	8	8	0.2188
DAE_I3	0	0.184454E-08	6	6	0.0625
DAE_IN2	0	0.922447E-03	31	18	0.8594
DAE_IN2X	0	0.368609E-12	12	9	0.1250
DAE_SYS	0	0.734715E-04	6	6	0.0156
DANWOOD	0	0.431731E-02	6	6	0.0000
DCA_CON	0	0.240842E-02	18	18	0.0000
DCA_HAR	0	0.288539E-02	20	20	0.0156
DCA_HYP	0	0.193146E-02	22	22	0.0000
DCIMMIGR	0	0.261171E-01	262	141	2.1875
DCKIN	0	0.291970E-04	228	121	0.9688
DCMDEG	0	0.248463E-02	29	29	0.8281
DEACT	0	0.433133E-03	44	34	0.3750
DEACTENZ	0	0.268680E-02	194	25	0.9844
DECAY	0	0.474818E-03	31	18	0.0781
DEFORM	0	0.323745E-03	36	22	0.2812
DEGEN	0	0.623480E-02	4	4	1.7500
DEGEN_M	0	0.818738E-01	5	4	1.8750
DEGRAD1	0	0.569791E+00	30	25	0.0469
DEGRAD2	0	0.408656E+00	38	29	0.1094
DEGRAD3	0	0.917811E-01	10	9	0.0000
DEGRAD4	0	0.110241E+00	17	17	0.0000
DEHYBENZ	0	0.986340E-04	22	17	0.0156
DENSITY	0	0.120623E-06	23	21	0.0156
DEWPOINT	0	0.100181E-03	85	19	0.2656
DFBLASER	0	0.572360E-12	17	17	0.0000
DFE1	0	0.105045E-03	455	216	0.0312
DFE2	0	0.368068E+00	7	7	0.0312
DIABETES	0	0.649514E-02	20	18	8.8594
DIAUXIA	0	0.264626E-02	31	27	3.2656
DIFDIST	0	0.143444E-03	7	7	0.0781
DIFFREAX	0	0.162774E-03	150	82	9.1875
DIFFUS_L	0	0.744400E-03	8	8	0.0781
DIMER	0	0.222450E+00	7	7	0.0156
DIODE	0	0.853611E-01	22	14	0.0625
DIS_KIN1	0	0.300643E-02	35	33	4.9531
DIS_KIN2	0	0.782674E-02	27	26	3.3750
DISLIQU	0	0.563701E-02	9	9	0.5000
DISORDER	0	0.478176E-01	78	17	1.0625
DISPLMNT	0	0.172442E-02	21	21	0.1562
DISRET_O	0	0.642544E-02	191	100	5.2031
DISS_ENZ	0	0.407355E-02	118	62	0.1094
DISS_TAB	0	0.350630E-02	70	44	0.0000
DISSOC	0	0.173006E-03	44	39	0.3125
DISTILL	0	0.742741E-04	37	36	47.6250
ETHYL	0	0.611772E-02	233	163	441.4219
ETHYL1	0	0.608413E-04	16	14	18.8750
ETHYL2	0	0.451887E-02	13	8	17.0156
EVAPOR	0	0.565781E-01	11	11	0.1406
EW_WAVEX	0	0.317667E-03	23	20	0.0156
EX_BREAK	0	0.455626E-10	93	61	0.0625
EXO_REAC	0	0.152273E-02	23	23	1.5781
EXOBATCH	0	0.345734E-02	119	18	2.5000
EXOTHERM	0	0.117690E-02	11	11	0.1875
EXP_FIT1	0	0.530658E-03	23	18	0.0156

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
EXP_FIT2	0	0.845454E-08	62	56	0.0312
EXP_FIT3	0	0.887315E-01	31	22	0.0000
EXP_FIT4	0	0.242852E-03	40	33	0.0000
EXP_FIT5	0	0.763338E-01	223	127	0.0625
EXP_FIT6	0	0.649129E-05	29	19	0.0000
EXP_FIT7	0	0.716406E-09	81	35	0.0000
EXP_INC	0	0.284761E+00	20	18	0.3281
EXP_P1	0	0.818151E-10	9	9	0.0000
EXP_P2	0	0.474473E-14	13	13	0.0000
EXP_P4	0	0.106701E-11	7	7	0.0000
EXP_P5	0	0.349077E-14	9	9	0.0000
EXP_P6	0	0.816621E-12	8	8	0.0000
EXP_P7	0	0.147492-151	16	9	0.0000
EXP_SIN	0	0.468463E+01	6	6	0.0156
EXP_SMPL	0	0.990021E-13	18	16	0.0469
EXP_SOL	0	0.730400E-03	12	12	0.0156
EXP_TEST	0	0.151119E+00	38	27	1.0000
EXP_TST	0	0.631447E-10	12	10	0.0000
EXP2TERM	0	0.189256E+02	20	15	0.0156
F_DEHYDE	0	0.885008E+01	258	100	59.1875
FAST	0	0.757590E-03	51	51	0.0938
FBR	0	0.200292E-02	63	53	0.4844
FC_EVAP	0	0.242397E-02	48	29	0.0469
FED_BAT	0	0.569246E+01	29	22	0.0469
FED_BATE	0	0.904511E-03	27	27	0.2031
FED10	0	0.268148E-02	73	51	1.8438
FEDBAT	0	0.160993E-02	59	41	5.1875
FEDBATCH	0	0.420249E+00	82	46	24.1250
FERMENT	0	0.588535E-02	39	27	0.2500
FERMNT	0	0.185162E-01	146	26	0.8125
FERMPROC	0	0.248175E-02	37	24	0.5938
FERMTEMP	0	0.141420E-03	33	27	0.4531
FIN	0	0.119747E-02	58	22	0.0000
FISH_POP	0	0.204687E-03	113	55	7.4219
FLUID_CL	0	0.127368E-02	11	10	0.0156
FLUOR	0	0.221858E-03	160	102	8.8438
FLUORES	0	0.477361E-03	24	19	0.4375
FLUORESC	0	0.974951E-03	30	25	18.0469
FOLDING1	0	0.193345E+00	15	15	0.0938
FOLDING2	0	0.102958E-02	65	50	0.2969
FOLDING3	0	0.103396E-03	41	30	0.0781
FOLDING4	0	0.223439E+00	44	13	0.1406
FOLDING5	0	0.744747E-01	10	9	0.0312
FOREST	0	0.153631E-02	600	61	2.8906
FRACTAK	0	0.154926E-01	151	35	0.1250
FUNGI	0	0.313665E-02	17	13	0.1094
FUNGI_I	0	0.105714E-02	31	25	0.0156
FUP_OSCI	0	0.144210E-02	62	54	15.3125
GAMMAS	0	0.794606E-03	52	52	0.0000
GAS_ABS1	0	0.176577E-02	7	7	0.1250
GAS_ABS2	0	0.181163E-02	6	6	0.7656
GAS_OIL	0	0.144212E-02	14	14	0.0312
GASCLOUD	0	0.459252E-10	10	9	0.0156
GASLIQ1	0	0.620602E-04	72	47	0.0938
GASLIQ2	0	0.217970E-02	51	45	0.1250
GAUSS	0	0.175297E-05	342	228	88.2188
GAUSS_3D	0	0.204785E-07	6	6	0.0000
GAUSS_AR	0	0.378974E-09	11	8	0.9688

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
GAUSS1	0	0.131582E+04	7	7	0.5000
GAUSS2	0	0.124753E+04	7	7	0.5312
GAUSS3	0	0.124448E+04	9	9	0.6719
GEAR	0	0.746704E-09	101	62	0.0781
GEO_PROB	0	0.780102E+04	15	11	0.0000
GIBBFUJI	0	0.151411E+00	95	64	17.3281
GLIDER	0	0.130862E-03	9	9	1.5625
GLOBCO2	0	0.633256E-02	18	15	0.3594
GLU_RATE	0	0.621973E-06	441	198	0.0312
GLUCOSE	0	0.965615E-03	153	86	1.4531
GLUCOSE1	0	0.433962E-02	3498	200	6.6406
GLUCOSE2	0	0.271563E+00	64	58	3.8281
GO1	0	0.224633E-14	16	15	0.0000
GO2	0	0.512318E-09	11	10	0.0000
GO3	0	0.791437E-25	8	8	0.0000
GO4	0	0.000000E+00	14	10	0.0000
GO5	0	0.512249E+01	7	7	0.0000
GO6	0	0.000000E+00	13	13	0.0000
GOLF	0	0.238683E-02	6	6	0.2969
GREASE	0	0.227485E-01	76	47	2.0625
GROWTH_H	0	0.125345E-01	15	13	0.0469
GYROS	0	0.143235E-03	146	75	2.9062
GYROSCOP	0	0.422938E-01	137	44	0.2656
HAHN1	0	0.276066E-04	48	30	2.0000
HAMILTON	0	0.232942E-18	7	6	0.0000
HEAT_XX	0	0.191480E-15	56	38	0.1719
HEATEX	0	0.299537E-03	121	42	2.9688
HEMISPH	0	0.377850E+05	9	9	0.0469
HEXA	0	0.165173E-03	22	22	0.3438
HIGH_ORD	0	0.859330E-23	3	3	0.0000
HIMMELBD	0	0.592256E+01	12	10	0.0000
HIRES	0	0.170912E-02	87	53	15.5781
HIRES_PA	0	0.164307E-02	281	145	39.0000
HIRUDIN	0	0.140588E-02	47	37	1.9062
HIV	0	0.214556E-02	8	8	0.0156
HMT	0	0.115534E-02	17	17	0.1094
HOLD	0	0.688293E-04	158	91	1.7500
HOLDUP	0	0.129998E-03	23	23	0.1875
HOLE	0	0.787378E-03	11	11	0.0312
HOMPOLY	0	0.388745E-04	16	16	0.0156
HYDENZYM	0	0.151118E-06	27	21	0.7344
HYDRODYN	0	0.324969E-01	6	6	0.0469
HYDROL	0	0.471349E-03	7	7	0.0312
HYDROXY	0	0.428475E-03	273	144	2.5000
HYP_ELA	0	0.135894E+00	10	10	0.5938
IDENT1	0	0.359714E-10	34	34	0.1562
IDENT2	0	0.187485E-04	34	34	0.0469
ILL_COND	0	0.118425E-14	5	5	0.0938
IMPULSE	0	0.373822E-01	6	6	0.0156
INC_STIF	0	0.348560E-02	20	17	0.0625
INFINITE	0	0.431831E-15	18	18	0.0156
INHIB	0	0.852271E-03	68	50	1.5156
INTEG_X	0	0.424276E-03	10	9	0.0000
INTERLEU	0	0.300411E-01	58	55	8.8281
INTPOL	0	0.349146E-10	11	11	0.0000
IO3EXP	0	0.149815E-01	46	36	1.2969
IONTRAN2	0	0.500904E-03	19	19	2.1562
IRB6400	0	0.187785E-10	58	32	7.3281

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
IRON_CC1	0	0.343561E-01	14	14	0.0000
IRON_CCH	0	0.826878E-02	41	11	0.0000
ISO_2PHA	0	0.240603E-02	10	10	0.0781
ISO_BAT	0	0.306048E+01	17	12	0.0312
ISOMER	0	0.153845E+02	36	12	0.3906
ISOMER_X	0	0.106061E+00	6	6	0.0000
ISOTOP1	0	0.510767E-01	229	120	48.8750
ISOTOP2	0	0.659771E-04	89	66	10.2500
JFIT	0	0.387546E-04	7	7	0.0156
JUXTA	0	0.927482E-02	28	23	0.0938
KATALY1	0	0.273798E+00	23	18	1.5469
KATALY2	0	0.810640E-01	43	42	2.1719
KEPLER	0	0.200615E+01	111	50	6.5312
KIDNEY	0	0.448472E+01	14	13	0.7031
KIN_PRO	0	0.290414E+00	107	65	2.3594
KINMOD	0	0.986794E-03	5	5	0.0781
KIRBY2	0	0.796549E-05	8	8	0.1562
KLADYN	0	0.355615E-02	36	29	1.0312
KNEE	0	0.154900E-02	69	39	0.0469
KS_FUNC	0	0.204684E-04	20	15	4.9062
LANCZOS1	0	0.262706E-16	1321	505	0.1250
LANCZOS2	0	0.115721E-11	1317	503	0.0938
LANCZOS3	0	0.836026E-09	1317	505	0.1094
LANDING	0	0.162521E-02	11	11	0.5312
LASER	0	0.177475E-02	23	18	0.2188
LEG_POL	0	0.190340E-10	7	7	0.0000
LEPS	0	0.713767E-02	84	48	10.2188
LIFE	0	0.194775E-02	9	9	0.0000
LIN_CMP1	0	0.574089E-06	102	67	0.0312
LIN_CMP2	0	0.372238E-02	31	30	0.0625
LIN_CMP3	0	0.276575E-01	126	68	0.0156
LIN_HC_X	0	0.945213E-03	25	24	0.5156
LIN_KIN	0	0.136505E-02	70	62	0.0312
LIN_MOD	0	0.618573E+00	12	10	0.0000
LIN_SYS	0	0.123755E-01	3	3	0.1406
LIN_VIS	0	0.215000E+00	296	187	1.6406
LINEWEAV	0	0.376989E+03	121	23	0.0156
LIQUID	0	0.218779E+00	658	250	0.1250
LISA	0	0.143201E+00	58	58	11.4219
LKIN	0	0.923705E-02	10	10	0.0469
LKIN_A1	0	0.923702E-02	31	23	0.0312
LKIN_A2	0	0.923703E-02	41	27	0.0781
LKIN_A3	0	0.922987E-02	44	36	0.0781
LKIN_A4	0	0.923702E-02	11	11	0.0312
LKIN_BR	0	0.659771E-03	87	53	0.3594
LKIN_L	0	0.935381E-02	8	8	0.0000
LKIN_L3	0	0.152552E-02	9	9	0.0312
LKIN_LA	0	0.984434E-03	134	79	1.3594
LKIN_NUM	0	0.923703E-02	12	12	0.0469
LKIN_O3	0	0.168327E-02	11	11	0.0312
LKIN_RE	0	0.425446E+03	42	11	0.0469
LKIN_S	0	0.923702E-02	110	71	0.6875
LKIN_T	0	0.923702E-02	113	73	0.2812
LKIN_X	0	0.199670E+03	7	7	0.0156
LKIN_X3	0	0.138290E+03	45	26	0.0781
LOC_EQUI	0	0.823932E-04	46	32	0.0938
LOG_GROW	0	0.247344E-01	7	7	0.0312
LORENZ	0	0.117320E-02	20	20	0.1875

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
LORENZ_S	0	0.239008E+03	14	14	31.3125
LOT_VOL1	0	0.328805E+00	17	15	3.4375
LOT_VOL2	0	0.275378E-01	11	11	0.0938
LYMPHO	0	0.212505E-02	54	36	7.0469
MAC_ECO	0	0.102054E-03	270	116	3.7500
MARBLE	0	0.155701E-02	8	8	0.7969
MARINE	0	0.193208E-01	25	19	1.0625
MARKET	0	0.194655E-05	189	105	0.7188
MARKET_C	0	0.108510E-02	7	6	0.0156
MCSTILL	0	0.345407E-03	159	100	15.3438
MCSTILLX	0	0.621918E+02	26	20	57.5156
MD_EQUI	0	0.234324E-03	8	7	0.0156
MDT_EQUI	0	0.134352E+01	307	67	0.0625
MECH_SYS	0	0.486904E+00	34	23	18.7812
MEM_WIRE	0	0.114346E-01	23	23	2.5781
MEMINH	0	0.561517E-03	20	19	0.4062
MEMSEP	0	0.242246E-02	14	14	0.2969
MENDES	0	0.535289E-01	136	123	277.5625
MET_SURF	0	0.132545E+00	111	67	4.6719
METHAN	0	0.333635E-01	21	21	0.0781
METHANE	0	0.220070E-02	140	39	0.0781
METHYL	0	0.199683E-02	5	5	0.0000
MGH09	0	0.207196E-02	835	325	0.0156
MGH10	0	0.226037E-07	621	222	0.0312
MGH17	0	0.382571E-05	22	18	0.0156
MICGROW	0	0.250562E-02	18	18	0.1719
MICGROWX	0	0.265791E+05	19	7	0.6719
MICGROWY	0	0.428623E+05	37	28	384.8594
MICGROWZ	0	0.116420E+04	61	2	75.6875
MICHMENT	0	0.195301E-01	8	8	0.0000
MILK1	0	0.572672E+00	61	50	3.4844
MILK2	0	0.625251E+00	120	73	3.6719
MIME	0	0.190356E-02	445	258	7.8594
MINWORLD	0	0.212429E-02	20	20	0.3125
MISRA1A	0	0.376748E-05	10	9	0.0000
MISRA1B	0	0.228268E-05	37	21	0.0000
MISRA1C	0	0.123918E-05	7	6	0.0000
MISRA1D	0	0.170659E-05	43	24	0.0156
MIX_PAT1	0	0.383537E-02	18	18	0.0156
MIX_PAT2	0	0.120451E-01	50	40	0.0312
MIX_PAT3	0	0.411979E-04	13	13	0.0156
MIX_PAT4	0	0.692848E-02	16	16	0.0156
MIX_RAT1	0	0.169166E-01	11	11	0.0156
MIX_RAT2	0	0.257307E-02	21	21	0.6875
MIX_RAT3	0	0.643448E-02	29	29	0.5156
MIXPOP	0	0.103448E-03	19	16	1.0469
MM_META1	0	0.137700E-11	11	11	0.0469
MM_META2	0	0.683845E-11	3	3	0.0156
MMKINET	0	0.662284E-03	53	37	0.0781
MN_CTRL	0	0.630305E+01	8	8	0.2812
MOISTURE	0	0.763523E-03	5	5	0.0156
MONOD	0	0.807830E+00	16	13	0.0000
MOON	0	0.828825E-01	5	5	0.0156
MORTALTY	0	0.617356E-02	12	12	0.0000
MOT_TSP	0	0.588072E-04	160	99	9.3594
MOTION	0	0.137359E-02	15	15	0.3906
MUBATCH	0	0.199568E-02	8	8	0.3594
MULT_CST	0	0.788967E-03	10	10	0.0312

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
MULTILAY	0	0.398470E-03	60	43	8.7188
MUSCLE	0	0.186980E-02	31	30	17.2500
MYL_ESTR	0	0.316189E-02	75	49	6.0312
NA_CSTR	0	0.115443E-02	45	38	0.0312
NC_RHS	0	0.425870E+00	35	35	3.5000
NELSON	0	0.154422E-01	234	125	1.7031
NITRIF	0	0.190024E-02	49	42	0.3750
NITRO	0	0.132678E-03	53	36	0.1250
NITROGEN	0	0.112542E+01	11	10	0.0312
NL_CSTR	0	0.648695E-02	65	65	2.2969
NL_ODE	0	0.153680E-02	66	19	0.0781
NLIN_VI1	0	0.293223E-02	15	15	2.8594
NLIN_VI2	0	0.281611E-02	43	34	134.8125
NLP_E5	0	0.165931E+06	110	31	0.0312
NLP_L5	0	0.288971E+01	15	10	0.0781
NLP_P1	0	0.533469E+01	9	9	0.0000
NLP_P3	0	0.164872E+01	2	2	0.0000
NLP_U3	0	0.111642E+00	57	39	2.9219
NON_DIFF	0	0.146531E-09	106	106	0.2344
NON_ISO	0	0.181830E-02	31	22	0.0625
NON_KIN	0	0.164384E-02	28	20	0.0938
NOSTR	0	0.183534E-02	33	24	0.0938
NTA1	0	0.337121E-01	86	51	6.4062
NTA2	0	0.109638E+00	37	21	1.6562
NUTRITI	0	0.433959E-04	33	29	0.8594
OAT1	0	0.152178E-02	10	10	0.0000
OAT2	0	0.736848E+01	18	17	0.0000
OBSERV1	0	0.111878E-02	10	10	0.0000
OBSERV2	0	0.347038E-02	32	20	0.1875
OBST_CTL	0	0.227936E+03	18	12	7.8438
OC_EX3	0	0.680946E-01	46	30	0.9844
OC_EX4	0	0.580331E+02	6	6	0.0000
OEKOSYS	0	0.153937E-02	18	14	0.0312
OIL	0	0.660959E-04	42	42	0.9531
OLIGO	0	0.258434E-02	127	91	3.7656
ON_OFF1	0	0.670554E-03	22	21	0.0312
ON_OFF2	0	0.234508E-02	2	2	0.0312
ON_OFF3	0	0.972139E-03	14	14	0.0781
ON_OFF4	0	0.713477E-02	28	26	3.6250
ON_OFF5	0	0.327245E-02	12	12	0.0312
ON_OFF6	0	0.105214E-02	13	13	0.0938
ON_OFF7	0	0.400251E-03	36	36	0.1406
OPT_CON	0	0.119362E+00	75	74	264.0625
OPT_CONT	0	0.103550E-02	86	47	0.2812
OPT_CTRL	0	0.167765E+03	61	52	0.6250
OPT_KIN	0	0.158372E+02	15	13	0.0469
OPT_KINX	0	0.158372E+02	16	14	0.0312
ORB_MOTN	0	0.198299E-02	117	72	0.8750
ORBIT	0	0.356695E-04	4	4	0.1719
OREGO	0	0.536913E+00	35	29	0.4375
OSC_REAC	0	0.538640E-03	40	32	1.8125
OSC_TRAN	0	0.764072E-01	15	14	4.3438
OSC2INTM	0	0.315115E-02	71	37	14.1094
OSCIL	0	0.135754E+00	52	35	1.7500
OSCILL	0	0.127426E-02	6	6	0.2188
OSCILL_S	0	0.143121E-17	586	262	0.4688
OSCILL_X	0	0.628739E-03	161	149	0.2500
OXDYN	0	0.828449E-03	22	18	0.0469

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
OXENZ	0	0.304259E-04	29	28	0.0625
OXIDAT	0	0.234104E-02	756	324	4.1562
OZONE	0	0.134857E-02	5	5	0.6562
P_IDENT	0	0.115298E-01	91	61	4.4219
P_IDENT1	0	0.853878E+00	65	44	8.1719
P_IDENT2	0	0.760548E-03	17	17	0.1875
P_IDENT3	0	0.862522E+12	9	8	0.3281
PAR_SIZE	0	0.192569E-02	17	17	148.2188
PARID120	0	0.730419E-01	22	15	0.1406
PARID15	0	0.285933E-02	19	16	0.0000
PARID30	0	0.511418E-02	8	8	0.0000
PARID60	0	0.471141E-01	26	17	0.0156
PARTICLE	0	0.969197E-12	5	5	0.0312
PCB	0	0.432160E+00	106	65	7.8750
PEAKS	0	0.101934E+01	18	13	0.4688
PECAN	0	0.826027E-02	10	10	0.2188
PEND_ELA	0	0.126012E-06	28	19	2.0156
PENDU_CS	0	0.268694E-02	33	15	0.8906
PENDU_I0	0	0.268582E-02	5	5	0.5469
PENDU_I1	0	0.270110E-02	5	5	0.0469
PENDU_I2	0	0.267295E-02	5	5	0.0469
PENDU_I3	0	0.126298E-03	27	4	0.1719
PENDU_IV	0	0.161585E-01	7	7	0.1875
PENDU_PD	0	0.268795E-02	44	12	0.7500
PENDULUM	0	0.125341E-03	6	6	0.3906
PENICILL	0	0.296841E-02	102	60	1.3906
PERIA	0	0.661274E-03	35	24	0.1719
PERM1	0	0.318054E-13	436	193	0.0000
PERM2	0	0.638663E-02	15	14	0.0000
PERM3	0	0.119491E+01	116	66	0.0156
PESTICID	0	0.105983E+02	10	10	0.0625
PHA_DYN1	0	0.250042E-03	12	12	3.5625
PHA_DYN2	0	0.126764E-03	30	30	10.7344
PHA_DYN3	0	0.731038E-03	8	8	0.5625
PHA_DYN4	0	0.717477E-03	10	10	1.4062
PHA_DYN5	0	0.130084E-02	82	55	1.4688
PHA_DYN6	0	0.126427E-02	82	55	0.4375
PHA_DYN7	0	0.246608E-03	75	47	0.3281
PHA_KIN1	0	0.730653E-02	77	47	0.0625
PHA_KIN2	0	0.165173E-02	37	30	0.3594
PHA_REAC	0	0.114356E-04	12	12	0.4531
PHARM_AP	0	0.346531E-04	10	10	0.0625
PHARMA	0	0.295131E+00	81	81	0.5469
PHB	0	0.310170E-02	70	42	0.2656
PHOS_TRA	0	0.931176E-03	61	55	2.0469
PHOSPH_A	0	0.239071E-02	19	19	0.3438
PHOSPH_D	0	0.253205E-02	22	14	0.1094
PHOTO	0	0.894887E-04	115	91	9.1875
PHOTO_PR	0	0.610227E-03	47	29	0.0781
PHOTO_S	0	0.172448E-02	7	7	0.6562
PHOTOCON	0	0.127421E-01	53	43	6.8594
PLANT_GR	0	0.141001E-02	99	58	1.7344
PLASMID	0	0.161171E-02	20	13	6.3281
PLASTER1	0	0.310648E-03	228	105	0.1562
PLASTER2	0	0.631588E-01	172	88	0.0469
PLASTER3	0	0.701560E-02	46	28	0.0156
PLASTER4	0	0.194008E-01	14	13	0.0156
PLATINUM	0	0.113068E+01	59	51	2.0625

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
PLUG_FLO	0	0.497949E-04	101	70	0.2500
POL_APP	0	0.246226E-03	38	8	0.0156
POL_MOD	0	0.485152E+00	10	10	0.0000
POLARI	0	0.153440E+00	153	36	3.9375
POLLUTNT	0	0.138657E-02	35	35	0.2344
POLY1	0	0.854521E-03	552	233	20.1094
POLY2	0	0.400926E-04	108	60	1.1719
POLYBU	0	0.107483E-02	20	20	0.7500
POLYBU_X	0	0.271382E-02	10	10	2.2656
POLYMER	0	0.110368E-02	253	127	2.5938
POPUL	0	0.924303E-03	46	37	2.0938
PROTEIN	0	0.222828E-02	29	22	0.6719
PROTOZOA	0	0.465926E-03	68	46	0.0156
PSS	0	0.603453E-01	38	38	0.0000
PYRIDIN	0	0.974355E-03	69	62	0.7188
PYROLYS	0	0.807628E-03	1384	500	127.5000
QUINIDIN	0	0.714700E+01	47	27	0.0000
RABBIT	0	0.126700E-01	19	19	0.2812
RAD_TRAC	0	0.125022E-01	87	57	0.0156
RAMAN	0	0.234515E-02	11	9	1.0938
RAMP	0	0.230151E-02	443	175	10.4375
RAT_APP	0	0.412971E-03	10	9	0.0000
RAT_FIT	0	0.307486E-03	47	35	0.0000
RAT42	0	0.442097E-03	14	13	0.0000
RAT43	0	0.233424E-02	47	33	0.0000
RATE_MOD	0	0.211498E-01	34	27	0.1562
RATSOL1	0	0.777047E-03	36	23	0.2344
RATSOL2	0	0.913270E-03	31	20	0.0781
RC_CAT	0	0.562323E-03	23	19	0.0625
RD_CSTR	0	0.407224E+00	169	57	7.8125
RE_ENTRY	0	0.978897E-03	71	40	1.3750
REAC	0	0.304242E-02	26	16	0.2656
REAC_CTR	0	0.492611E-09	42	23	1.0781
REACMECH	0	0.346189E-02	7	7	0.2344
REACTION	0	0.367981E+00	10	9	0.0469
RECLIC19	0	0.197299E-03	12	10	0.0000
RECLIG1	0	0.320021E-02	34	30	0.0312
RECLIG10	0	0.450560E-03	28	19	0.0156
RECLIG11	0	0.405838E+02	10	10	0.0156
RECLIG12	0	0.110311E+01	12	12	0.0156
RECLIG13	0	0.164165E+03	15	15	0.0000
RECLIG14	0	0.136664E+00	33	33	0.0625
RECLIG15	0	0.140112E-05	78	48	0.0625
RECLIG16	0	0.287350E-03	105	53	0.0469
RECLIG17	0	0.738716E-06	264	126	0.0469
RECLIG18	0	0.211688E-02	983	473	0.4531
RECLIG2	0	0.219188E-02	55	31	0.0469
RECLIG3	0	0.228106E-02	43	37	0.0156
RECLIG4	0	0.192291E-02	9	9	0.0469
RECLIG5	0	0.239576E-03	111	61	0.0312
RECLIG6	0	0.648014E-03	86	59	0.0469
RECLIG7	0	0.203802E-02	6	6	0.0312
RECLIG8	0	0.250929E-02	25	19	0.0156
RECLIG9	0	0.547390E-05	8	8	0.0156
REFLECT	0	0.202089E+01	175	109	0.0781
REFLECTY	0	0.169970E-01	6	6	0.0625
REFRIG	0	0.162665E-02	108	35	0.1406
REG_RES	0	0.257572E-02	44	34	1.0000

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
RELAY	0	0.264765E-04	15	14	0.4531
RELEASE	0	0.202554E-03	13	13	0.0312
REPFED	0	0.581629E-03	81	46	0.3125
REPLCUL	0	0.206763E-03	8	8	0.2031
RES_TIME	0	0.164185E-16	5	5	0.0000
RESPIR	0	0.185873E-02	4	4	0.0000
RESPIRA	0	0.384931E+01	10	10	0.0781
REVTEMP	0	0.293166E-02	68	41	0.6250
REXT	0	0.751967E-03	28	22	0.7344
RICH_GR	0	0.131655E-02	352	177	0.0156
ROB_ARM	0	0.314159E-02	6	6	0.0312
ROB_CTRL	0	0.904516E-03	16	14	1.4375
ROBERT	0	0.172651E-02	8	8	0.0469
ROBOT	0	0.353425E-02	6	6	1.0312
ROBOT_2	0	0.392925E+00	2331	166	51.5156
RODC	0	0.382995E-03	12	12	0.0156
ROESSLER	0	0.359659E-04	42	35	0.1562
ROSZMAN1	0	0.104282E-03	6	6	0.0156
RT_PULSE	0	0.279726E+02	103	42	0.1875
RTD	0	0.144962E-04	18	18	0.0000
RUN	0	0.379550E-03	32	17	0.1406
SAT_EXP	0	0.245649E-02	40	35	0.1875
SBR_COOL	0	0.345608E-02	26	19	1.2969
SCARA	0	0.987136E-02	7	7	1.4531
SCARA_X	0	0.171291E+06	20	20	48.0781
SCP	0	0.125826E-01	37	29	1.7031
SE	0	0.199671E+00	46	38	0.0625
SEIR	0	0.271837E-02	27	25	0.6719
SEMIPAR	0	0.871584E-03	8	8	0.1719
SEMISEG	0	0.875836E-03	25	23	0.0938
SEMISEQ	0	0.108617E-02	8	8	0.0312
SENS	0	0.129308E-02	43	27	0.1562
SEQ_EXP	0	0.893317E-02	15	15	0.0000
SHAKER	0	0.154325E-02	22	16	0.1406
SHARP1	0	0.800224E-04	8	8	0.0469
SHARP2	0	0.356558E-01	132	59	0.1406
SHEEP	0	0.317093E-02	10	10	0.2812
SHELL	0	0.100115E-02	38	38	0.6250
SHOCK	0	0.732841E-04	19	14	0.0781
SIMP_ECO	0	0.839739E-03	115	71	3.5781
SIMP_POP	0	0.513022E-02	7	7	0.0000
SING_EQU	0	0.695117E-02	7	7	0.0156
SKIN_D	0	0.181479E-01	124	79	6.7188
SLIP	0	0.273344E+00	330	37	136.8438
SLUDGE	0	0.860007E-04	259	163	231.2812
SMALL_EX	0	0.185309E-04	14	14	0.0000
SMOOTHNG	0	0.167405E-03	32	32	0.7188
SOOT	0	0.190921E-02	47	38	0.3438
SPBEDRTD	0	0.351033E-02	16	13	0.3906
SPLINE	0	0.526763E-20	3	3	0.0000
SPRING	0	0.443291E-10	37	24	0.1562
SS_REAC	0	0.101322E-02	15	13	0.2031
SS_TUBE	0	0.723365E-01	115	41	0.0469
SSHEATEX	0	0.697913E-13	20	19	0.0625
STABCSTR	0	0.145636E-03	42	31	0.4062
STABIL	0	0.321953E-04	9	9	0.0312
STAGED	0	0.135589E-02	30	22	0.2031
STAR	0	0.173121E+00	14	12	0.2344

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
STARS	0	0.442915E-01	10	6	1538.0625
STAT5	0	0.238896E-02	20	16	0.1875
STEP_RES	0	0.918965E+00	29	19	0.0000
STIFF	0	0.765200E-02	75	55	1.2344
STIFF_DE	0	0.277107E-02	15	14	0.0156
STIFF_EQ	0	0.173957E-02	50	30	0.2812
STIFF1	0	0.857372E-09	15	15	0.0000
STIFF2	0	0.882775E-16	104	83	2.5469
SUB_ACCU	0	0.284266E+01	14	13	0.0156
SUBTILIS	0	0.170916E-02	25	19	1.1094
SULFATE	0	0.594394E-02	14	14	0.0000
SULFUR	0	0.288922E+00	39	38	2.6562
SULPHUR	0	0.646683E-04	6	5	0.0625
SUPEROX1	0	0.700884E-02	16	15	0.2969
SUPEROX2	0	0.181756E-02	23	23	0.8281
TAB_DIS1	0	0.967347E-04	115	87	1.5000
TAB_DIS2	0	0.182185E-03	52	52	0.4844
TAB_DIS3	0	0.292362E-03	66	51	1.2812
TAB_DIS4	0	0.178015E-02	60	50	0.6406
TAB_DIS5	0	0.757277E-03	27	27	0.6406
TAB_DIS6	0	0.158980E-02	29	24	0.4219
TAB_DIS7	0	0.315078E-02	1348	500	14.5312
TAB_DIS8	0	0.904915E-02	333	186	21.6875
TANK	0	0.313506E-04	16	16	0.0469
TANKBLD	0	0.474984E-03	15	13	0.0000
TANKDIS	0	0.216254E-03	15	14	0.0156
TANKHYD	0	0.569069E-04	71	50	0.3906
TEFLON1	0	0.245480E-03	183	102	3.1250
TEFLON2	0	0.347330E-03	166	93	48.8438
TEMP_LEV	0	0.108249E-03	3	3	0.0156
TEMPCONT	0	0.668493E-03	21	21	0.0312
TEST_0DE	0	0.577450E-03	11	11	0.0156
THERM	0	0.146178E-02	26	23	0.1875
THERMRES	0	0.349926E-02	24	24	0.0000
THURBER	0	0.165175E-03	36	32	0.0156
TIME_ACT	0	0.107183E-02	93	58	0.0000
TITRATIO	0	0.931205E-04	12	12	0.3438
TOLUENE	0	0.674171E-01	207	114	5.1094
TOPTCSTR	0	0.290255E-03	26	26	2.0312
TP1	0	0.272049E-15	7	7	0.0000
TP1_A	0	0.105326E-22	8	6	0.0000
TP1_B	0	0.493143E-19	8	6	0.0000
TP13	0	0.100203E+01	18	18	0.0000
TP14	0	0.139346E+01	37	7	0.0000
TP2	0	0.494123E+01	7	7	0.0000
TP202	0	0.489843E+02	12	10	0.0000
TP203	0	0.112089E-15	7	7	0.0000
TP205	0	0.656250E+00	3	3	0.0000
TP212	0	0.726440E-18	4	4	0.0000
TP241	0	0.434229E-16	17	14	0.0000
TP242	0	0.288413E-14	11	11	0.0000
TP244	0	0.161527E-13	9	9	0.0156
TP246	0	0.252682E-29	4	4	0.0000
TP247	0	0.810000E+00	26	19	0.0000
TP25	0	0.443495E-15	229	103	0.5469
TP256	0	0.550911E-18	18	18	0.0000
TP260	0	0.670781E-19	58	38	0.0000
TP261	0	0.264156E-17	67	61	0.0156

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
TP267	0	0.363560E-18	29	27	0.0000
TP269	0	0.409302E+01	3	3	0.0000
TP272	0	0.323476E-10	349	147	0.0156
TP282	0	0.129053E-14	77	43	0.0156
TP286	0	0.160124E-13	63	37	0.0156
TP288	0	0.435622E-17	16	16	0.0000
TP303	0	0.469943E-21	13	11	0.0000
TP307	0	0.124362E+03	26	19	0.0000
TP308	0	0.773199E+00	52	19	0.0000
TP312	0	0.592256E+01	45	13	0.0156
TP327	0	0.284597E-01	9	9	0.0000
TP332	0	0.115442E+03	85	46	1.8125
TP333	0	0.432704E-01	9	9	0.0000
TP334	0	0.538115E-01	8	8	0.0000
TP350	0	0.244613E-03	29	27	0.0156
TP351	0	0.318572E-01	37	22	0.0000
TP352	0	0.903234E+03	130	8	0.0000
TP354	0	0.113784E+00	11	10	0.0000
TP355	0	0.622512E+01	158	82	0.0156
TP358	0	0.408050E-04	5	5	0.0000
TP370	0	0.227419E-02	9	8	0.0469
TP371	0	0.139823E-05	7	6	0.0312
TP372	0	0.133901E+05	6	6	0.0000
TP373	0	0.133901E+05	40	8	0.0000
TP379	0	0.401377E-01	32	22	0.0781
TP394	0	0.191667E+01	22	22	0.0469
TP43	0	0.956000E+03	30	18	0.0000
TP46	0	0.148387E-13	20	20	0.0156
TP48	0	0.479889E-09	7	7	0.0000
TP57	0	0.284597E-01	45	9	0.0000
TP6	0	0.214111E-17	27	16	0.0000
TP70	0	0.149963E-02	31	28	0.0156
TPGLOB	0	0.114324E+01	16	14	0.0000
TRAJ_NET	0	0.732153E+03	203	20	0.0156
TRAN_A_B	0	0.781469E-04	16	11	0.1094
TRANS90	0	0.138102E-03	9	9	0.0000
TRANS90X	0	0.117899E-04	2	2	0.0000
TRANSIST	0	0.467351E-03	35	26	0.8125
TRANSUS1	0	0.120872E-01	76	34	0.7344
TRANSUS2	0	0.405819E-01	174	109	6.6250
TRAY	0	0.571405E-01	23	23	0.0312
TREES	0	0.488210E+01	34	23	3.6875
TREND	0	0.386306E-04	19	14	10.3125
TRICHO	0	0.171646E-02	30	24	0.1250
TRIG_APP	0	0.109517E-01	7	7	0.0000
TRILEPT	0	0.157144E-02	25	22	1.5312
TRUCK	0	0.561678E+00	23	20	5.5312
TUBDYN	0	0.568172E-02	95	55	4.1719
TUBED	0	0.358364E-02	37	23	0.0000
TUBEMIX	0	0.150253E-02	87	54	0.6406
TUBTANK	0	0.118457E-01	4	4	0.0000
TUBULAR	0	0.188550E-02	31	29	1.3906
TURBCON	0	0.257502E-02	11	11	0.1094
TWOEX	0	0.277405E-02	27	25	0.3906
TWOONE	0	0.161294E-02	23	17	0.6094
TWOSTAGE	0	0.140214E-02	29	19	0.0938
TWOTANK	0	0.649886E-04	17	17	1.9375
ULTRA1	0	0.167331E-01	6	6	0.0000

(continued)

name	i_f	$f(x^\star)$	n_f	n_{its}	*time*
ULTRA2	0	0.117433E-03	12	12	0.0000
UPTAKE	0	0.229463E+00	158	77	3.0781
URETHAN	0	0.641777E-02	40	27	23.9219
URETHANW	0	0.156446E+02	32	29	675.4844
URETHANX	0	0.569526E+02	3	3	110.6719
US_CSTR	0	0.129513E-01	9	9	0.0781
VAPOR	0	0.547108E-04	9	9	0.0156
VAR_META	0	0.119430E-01	187	132	67.0000
VARMOL	0	0.113218E-02	58	53	0.2969
VARVOL	0	0.186141E-02	26	19	0.7344
VAS_ADSP	0	0.503009E+00	125	37	0.3281
VAS_ADSS	0	0.251177E-01	313	173	22.1562
VDPOL	0	0.587628E-08	4	4	0.0000
VDPOL_0	0	0.294378E-10	10	10	0.0156
VDPOL_M	0	0.150981E-01	30	25	0.1406
VERT_CHL	0	0.564238E-03	22	21	0.4531
VESSEL	0	0.277916E+00	8	8	0.0000
VIS_SPEC	0	0.243749E-01	11	11	0.4844
VISC_ELA	0	0.977661E-01	58	47	0.0312
VISCREG1	0	0.719344E+00	21	11	0.0156
VISCREG2	0	0.111928E-03	11	9	0.0469
VISCREG3	0	0.514122E-01	49	48	0.2969
W_WATER1	0	0.857287E-02	22	18	0.0156
W_WATER2	0	0.412164E-03	307	175	3.6250
WASTEWAT	0	0.649384E-03	26	18	0.0781
WAVE_D1	0	0.343510E-02	4	4	0.0156
WAVE_D2	0	0.115289E-01	9	9	0.0625
WAVE_X	0	0.716972E-03	10	9	0.0156
WEEDS	0	0.219779E-01	38	38	0.0312
WEIBEL	0	0.134893E-02	7	7	0.0000
WEIBULL	0	0.101469E-02	9	9	0.0000
WNDSHEAR	0	0.114596E+04	12	11	4.0938
YEASTOSC	0	0.224576E+02	46	10	1.5469
ZOOPLANK	0	0.179688E-02	71	49	6.9688
ZS_EQTN	0	0.843541E-02	8	8	0.0000

Table 9: Individual Test Results for PDEFIT

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
2AREAS	0	0.197214E-01	82	36	6.2969
2AREAS_C	0	0.823845E-03	82	49	4.7031
2MEMBRAN	0	0.253409E-03	36	34	43.3906
2ND_DIR1	0	0.177277E-01	36	30	0.6406
2ND_DIR2	0	0.169355E-01	103	52	1.6406
ACCRET	0	0.145667E-01	7	5	0.1406
ACCRET_A	0	0.141639E-01	4	4	1.7656
ACCRET_F	0	0.169557E-02	31	21	56.2031
ACETYL_T	0	0.289513E-05	5	5	48.2344
ACETYL_Z	0	0.525728E-03	18	18	102.9531
ADV_DIFF	0	0.480205E+02	16	10	14.2656
ADV_DOM	0	0.905220E-02	91	50	13.1250
ADV_DOMS	0	0.156990E-03	23	19	22.3281
ADV_VC	0	0.394030E+01	7	7	2.3750
ADV_VTC	0	0.129787E-01	4	4	6.8750
ADVEC_2N	0	0.156495E-02	13	9	0.3281
ADVEC_5N	0	0.214269E-01	13	9	58.2344
ADVEC_CP	0	0.249513E+00	54	36	130.6094
ADVEC_LU	0	0.460407E-02	19	12	5.8125
ADVEC_PB	0	0.130745E+01	11	9	0.9062
ADVEC_S	0	0.889902E-03	711	100	31.9062
ADVECT	0	0.491682E+00	81	23	4.1094
ADVECT_N	0	0.801325E-01	24	15	2.0000
ADVECT_R	0	0.289361E+01	121	46	23.9219
ADVECT_S	0	0.801305E-01	12	11	5.0312
ADVECT2	0	0.291403E+01	39	22	13.9844
ADVECT2A	0	0.239003E+01	34	20	29.3906
AFFIN	0	0.736798E-02	18	18	3.3906
AIR_FLOW	0	0.289397E-03	17	7	287.0625
AL_ALLOY	0	0.401143E+01	26	16	0.2812
ALDRIN_P	0	0.467253E-01	52	28	4.9062
ARA_YARN	0	0.709130E-02	9	9	15.1562
AX_DIFF	0	0.156461E-05	16	12	1.0000
BALLONS5	0	0.312335E-01	44	10	11.3750
BEAM1	0	0.628481E-02	6	6	0.2344
BEAM2	0	0.885221E-02	6	6	0.2344
BEETLES	0	0.508943E-01	24	4	9.9688
BIFURC1	0	0.306036E-03	206	73	271.2031
BIFURC2	0	0.659065E-03	200	101	275.1719
BINDSITE	0	0.101903E+00	11	10	61.9062
BIOFILM	0	0.119588E-02	18	18	13.5469
BLD_BRN	0	0.299474E+00	42	30	10.0938
BLOW_UP	0	0.537306E-02	8	8	1.1875
BRAIN	0	0.736917E-02	23	18	1.6562
BRINE	0	0.150560E+00	8	6	54.9844
BRUSS2D	0	0.273659E+01	19	16	632.1250
BRUSSEL	0	0.772691E+00	130	70	1207.1406
BSE	0	0.242493E-03	50	29	4.1875
BUBB_BIO	0	0.935037E-03	8	8	0.6719
BUBBLE	0	0.279334E-02	10	9	1.8281
BURGER	0	0.139853E-02	11	11	3.0938
BURGER_E	0	0.657438E-04	85	45	7.7812
BURGER_F	0	0.293878E-01	7	7	0.3125
BURGER_I	0	0.110622E-01	177	52	370.0312

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
BURGER_R	0	0.776713E-02	80	51	116.2031
BURGER_X	0	0.146830E-01	116	62	25.5938
BURST	0	0.396054E-02	9	8	1.8125
BVP_TRIV	0	0.597932E-07	6	6	0.0000
CALIBR	0	0.817088E-04	229	50	6.6875
CAPILL	0	0.691220E-02	6	6	0.3438
CARRIER	0	0.982155E-03	22	14	0.3438
CD_TRANS	0	0.953159E-02	7	6	2.8906
CHRESIGN	0	0.216809E-02	18	18	847.9688
CN_PLAS	0	0.869394E-04	17	13	15.4375
CNT_CUR1	0	0.114271E-02	17	17	3.5000
CNT_CUR2	0	0.114260E-02	26	26	17.4219
CO_OXYD	0	0.254921E-02	61	40	75.9688
COLLAGEN	0	0.162207E-02	33	19	45.0625
COMP_MED	0	0.422863E-04	14	14	0.4219
CON_DIV1	0	0.680996E-09	7	7	0.3906
CON_DIV2	0	0.469667E-05	14	10	0.2188
CONTAMIN	0	0.252108E-03	8	8	0.5625
COS_PROF	0	0.833562E-02	42	25	44.4375
CPA_PLAS	0	0.199805E-02	21	21	1.2969
CPL_ADV	0	0.333861E-02	18	16	9.2188
CRYSTAL	0	0.198355E-02	8	8	0.8438
CSE	0	0.341350E+01	17	9	178.1562
CTFLOW	0	0.407426E-03	10	9	1.9219
CTFLOW_P	0	0.100885E-01	7	7	0.2344
CTRL_WAV	0	0.512221E-02	19	19	127.8750
CUBIC	0	0.925690E-03	22	18	6.3125
CUSP	0	0.213583E-02	10	8	151.0312
DAMBREAK	0	0.380106E-02	5	5	31.9531
DBVP	0	0.529041E-02	43	30	0.7656
DC_TUBE	0	0.277152E-03	29	18	0.3906
DEHYDRO	0	0.391910E-07	22	22	3.7812
DERMAL	0	0.128512E+00	110	51	7.8906
DESIGN	0	0.280434E-02	38	9	5.4844
DIALYSI1	0	0.100379E-04	38	23	1.9844
DIALYSI2	0	0.852832E-04	32	19	0.2812
DIALYSI3	0	0.328439E-05	110	59	1.5625
DIALYSI4	0	0.238740E-04	423	158	53.8906
DIALYSI5	0	0.591920E-04	92	44	9.5625
DIFF_1D	0	0.277129E-03	38	25	0.9531
DIFF_ADS	0	0.115340E+00	33	21	4.8594
DIFF_CON	0	0.954675E-02	8	8	0.2500
DIFF_ETH	0	0.432093E-02	7	7	0.7969
DIFF_NLB	0	0.183100E-01	32	24	6.9375
DIFF_P	0	0.705988E-01	34	20	1.2500
DIFFPT	0	0.503863E-05	7	7	0.2031
DIFFPT1	0	0.312793E-02	732	94	4.7812
DIFFPT2	0	0.503863E-05	7	7	0.1875
DIFFPT3	0	0.182061E-01	14	10	8.6406
DIFFPT4	0	0.135434E-01	15	15	7.8281
DIFFREA	0	0.251659E-02	16	8	2.0625
DIFFREA1	0	0.245002E-03	8	8	9.2656
DIFFUS	0	0.994658E-02	28	19	0.1719
DIGESTOR	0	0.302969E-02	11	11	3.0312
DISADVFL	0	0.294066E-02	11	11	2.6562
DISRE	0	0.128143E-01	20	17	12.0938
DISRET	0	0.143346E-01	39	28	7.6250
DRY	0	0.670730E-04	43	40	0.2812

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
ECOLOGY	0	0.219609E+01	132	65	54.5156
ELA_TUBE	0	0.592056E-02	89	48	16.8125
ELASTIC	0	0.500746E-02	16	12	222.6875
ELDYN_A	0	0.244715E-02	24	20	24.1406
ELEC_DYN	0	0.259297E-01	24	24	34.4531
ELECTRO	0	0.922633E-02	9	9	2.4531
ELLIPTIC	0	0.440546E-03	19	19	0.5625
ENERGY	0	0.133307E-06	11	10	0.1719
ENZDYN	0	0.165730E-02	15	14	1.8750
EW_WAVE	0	0.774823E+01	9	9	3.8750
EXR14_1	0	0.645539E-03	18	16	7.5312
EXR14_2	0	0.690227E-03	4	4	0.5000
EXR14_3	0	0.164123E-01	90	61	39.2500
EXR15_4	0	0.524956E-03	10	9	5.9688
EXR21_1	0	0.423643E-02	25	18	14.9844
EXR22_1	0	0.664933E-02	26	17	6.1406
EXR23_1	0	0.550607E-03	7	7	0.2188
EXR24_1	0	0.706416E-03	5	5	1.8906
EXR25_1	0	0.816060E-03	3	3	0.0156
EXR25_2	0	0.219549E-03	34	22	14.9844
EXR25_6	0	0.503566E-03	14	12	1.0781
EXR31_4	0	0.945764E-05	6	6	40.6250
EXR32_1	0	0.105541E-01	22	17	57.6562
EXR32_4	0	0.303460E-02	15	11	1.6875
EXR32_7	0	0.251976E-01	31	13	31.6562
EXR32_8	0	0.153593E-02	6	6	11.6875
EXR32_9	0	0.387870E-02	42	28	93.2344
EXR34_12	0	0.578055E-03	99	54	218.2344
EXR42_2	0	0.795779E-03	6	6	1.4219
EXR43_2	0	0.259086E-03	23	16	2.2812
EXR43_3	0	0.266703E-03	5	5	0.5312
EXR52_5	0	0.152086E-02	7	7	14.1250
EXR64_4	0	0.153021E-02	41	26	8.7031
EXR65_5	0	0.732749E-03	8	8	1.8125
FILTWASH	0	0.226865E-02	12	12	16.5156
FINAG	0	0.588375E-02	79	44	49.8594
FIX_BED	0	0.586050E-03	74	41	776.8125
FIXBED	0	0.373599E+02	41	30	16.7188
FLAME	0	0.107380E-02	11	11	10.7344
FLAT_MEM	0	0.274450E-01	144	75	8.9375
FLOW	0	0.163319E-02	7	7	1.3750
FLOW_PMD	0	0.215229E-03	4	4	0.4531
FLOW_PMW	0	0.702982E-02	9	9	3.3281
FLOWDIFF	0	0.849421E-03	9	9	40.3281
FLUID	0	0.762892E-03	216	89	187.5938
FOX	0	0.205531E-03	60	40	145.6719
FRONT	0	0.282696E+00	228	85	79.4688
G_HILL	0	0.995242E-04	43	26	7.5000
GAS_BUBB	0	0.976182E+00	3	3	0.1719
GAS_CONV	0	0.217208E-02	58	43	6.6094
GAS_DIF1	0	0.907982E-02	6	6	0.0625
GAS_DIF2	0	0.584899E+01	5	5	0.0312
GLACIER	0	0.330474E-02	9	9	7.2188
GLYCO	0	0.687518E-04	7	7	1.1406
GROUND_W	0	0.746352E-02	110	19	83.9062
GROWTH	0	0.548371E-02	12	11	0.2188
HEAT	0	0.111422E-03	4	4	0.0312
HEAT_A	0	0.162359E-02	5	5	0.1562

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
HEAT_B	0	0.213749E-01	41	23	2.7031
HEAT_BD3	0	0.131805E-02	13	12	1.1094
HEAT_CD	0	0.571725E-05	14	14	0.5156
HEAT_CF	0	0.588450E-03	77	55	4.8125
HEAT_CON	0	0.322209E-02	110	60	77.1562
HEAT_CW	0	0.605904E-02	57	37	0.4844
HEAT_CYL	0	0.386842E-02	16	15	0.4375
HEAT_EX	0	0.474901E-03	22	22	2.4531
HEAT_F	0	0.100741E-02	12	10	0.7188
HEAT_I	0	0.185764E-02	43	28	0.4219
HEAT_MS	0	0.337230E-11	6	6	0.0469
HEAT_NLB	0	0.471190E+00	14	13	0.8125
HEAT_NLC	0	0.480718E+00	7	6	0.0938
HEAT_NLD	0	0.471205E+00	13	13	1.3750
HEAT_R	0	0.226342E+00	5	5	1.3281
HEAT_RAD	0	0.787176E-03	7	7	1.5312
HEAT_S1	0	0.730107E-01	7	7	0.1562
HEAT_S2	0	0.831368E-02	9	8	0.7344
HEAT_SEN	0	0.236741E-02	18	12	2.5625
HEAT_SX	0	0.256629E-04	17	12	0.6875
HEAT_TDC	0	0.735501E-02	22	22	4.2031
HEAT_X	0	0.203687E-02	17	11	0.0938
HOT_SPOT	0	0.972604E-02	244	46	17.3906
HUMID	0	0.859300E-04	50	36	39.9062
HYDRO	0	0.278388E+00	26	17	34.2500
HYDRO_2C	0	0.313435E-02	12	10	66.9219
HYDRO_3S	0	0.691002E-01	79	12	723.7812
HYDRO_FX	0	0.279698E+00	6	6	16.2656
HYG_POLY	0	0.351034E-02	5	5	0.4375
HYGROS	0	0.763480E-02	67	17	0.3125
HYP_PBC	0	0.210677E-02	21	15	2.1406
HYP2ND	0	0.487786E-01	5	5	131.3906
HYPER	0	0.737105E-01	22	21	8.2031
HYPERBO1	0	0.838491E+00	7	7	0.3906
HYPERBO2	0	0.106866E+01	11	10	0.5469
HYPERBO3	0	0.107103E+01	13	12	0.9531
HYPERBO4	0	0.101498E+01	17	14	1.5938
HYPERBO5	0	0.817186E+00	13	12	1.1562
IN_LAYER	0	0.110752E-02	37	24	11.5625
INTEG	0	0.874436E-01	109	65	73.2656
INTERF1	0	0.473012E-02	9	9	0.2344
INTERF2	0	0.137618E-02	18	18	1.3594
INV_PROB	0	0.264149E-02	10	10	11.6562
IONTRAN1	0	0.323129E-04	24	15	0.7500
IONTRAN3	0	0.544073E-02	59	38	9.5000
ISOTHRM1	0	0.253721E-03	9	9	39.3594
ISOTHRM2	0	0.118388E-02	20	19	41.1719
JONTO	0	0.120297E+01	9	9	0.7500
KDV1	0	0.228367E-01	4	4	145.3594
KDVE	0	0.390824E-05	6	6	2.0625
KILN	0	0.174322E-05	14	14	2.7969
KIN_SORP	0	0.132174E-02	38	27	103.0781
LAM_FLOW	0	0.610134E-02	3	3	0.0312
LAPLACE	0	0.462665E-08	11	8	4.8125
LDCP	0	0.810827E-01	20	15	7.2656
LIN_ADV	0	0.101996E-02	5	5	3.7344
LIN_HC	0	0.381197E-06	57	40	23.7969
LIN_HYP1	0	0.125108E+01	12	10	2.3594

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
LIN_HYP2	0	0.181002E+01	54	32	7.2188
LIN_HYP3	0	0.195746E-01	201	37	15.3125
LNCHROM1	0	0.159458E-06	7	7	561.3594
LNCHROM2	0	0.180140E+00	7	7	5.4219
LOSSLESS	0	0.731487E-04	7	7	0.3438
LUNG	0	0.179731E-02	22	19	2.6562
MALTDX10	0	0.651074E+00	6	6	188.6562
MALTODEX	0	0.298032E-01	9	9	5.8125
MASS_TRA	0	0.693990E-02	8	8	0.0312
MC_DIST	0	0.245082E-01	15	14	11.1719
MCFC1	0	0.428140E-02	8	8	2345.0938
MCFC2	0	0.242082E+01	12	9	175.3594
MCFC3	0	0.365468E-03	11	11	1009.4844
MCFC_0	0	0.859807E-04	12	12	1531.7969
MEDAKZO	0	0.679731E-03	31	31	92.8906
MEM_SEP	0	0.369087E-02	48	33	12.0000
MEMBRANE	0	0.193512E-02	17	13	3.6875
MILL1	0	0.655771E-08	4	4	1.4219
MILL2	0	0.326179E-04	18	8	33.8750
MILL3	0	0.277054E-02	22	22	16.7812
MOL_DIFF	0	0.213780E-01	35	32	4.3594
MOVFRONT	0	0.791753E-02	31	23	4.7344
MX_ENTRO	0	0.459350E-02	93	56	164.7031
MZ_FURN	0	0.241212E-05	76	56	20.9844
N_CONVEX	0	0.266185E+00	3	3	0.1406
NDYN	0	0.127796E-01	49	20	1.8125
NERVE	0	0.231102E-01	452	200	1463.0781
NET_3	0	0.160726E+01	16	16	11.5156
NL_HEAT	0	0.102788E-02	23	20	5.6094
NL_PDE	0	0.477259E-09	7	6	0.2500
NL_PDE1	0	0.535725E-03	8	7	0.0781
NL_TRANS	0	0.508577E-02	4	4	1.7969
NL2_SORP	0	0.202880E-01	10	9	3.3750
NLIN_2ND	0	0.275586E-02	17	13	3.9062
NLINPDE	0	0.271982E-08	16	16	64.1094
NLSE	0	0.124533E-04	12	11	7.8438
NOISE	0	0.140632E-01	12	12	2.1719
NON_AD	0	0.156634E-01	4	4	0.2812
OBSTACLE	0	0.743737E-03	2	2	0.0469
ONESTEP	0	0.330138E-03	43	26	93.5000
OSC_SOL	0	0.750426E-03	10	10	2.6562
PACK_BED	0	0.447213E-01	9	9	0.8125
PAR_CTRL	0	0.184680E-11	16	12	0.2031
PAR_SIN	0	0.861061E-10	13	9	0.0625
PAR_SINA	0	0.569298E-08	6	6	0.5156
PARAB1	0	0.212411E-08	6	6	0.0625
PARAB2	0	0.313050E-09	9	9	0.4688
PARAB3	0	0.886363E-09	12	10	0.0781
PARAB4	0	0.180014E-05	11	11	0.3281
PARAB5	0	0.790570E-08	6	6	0.0781
PARAB6	0	0.712814E-06	7	7	0.2188
PB_CTRL1	0	0.247435E-02	36	36	217.0938
PB_CTRL2	0	0.664554E-02	27	23	20.3281
PDAE4	0	0.318025E-08	5	5	0.8594
PERIOCHP	0	0.564834E-02	19	19	72.7188
PHYP_PBC	0	0.122068E-02	26	18	1.8750
PLASMA	0	0.151578E-02	40	17	6.7656
POLLUTN	0	0.375137E-02	21	10	14.7188

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
POLY_DYN	0	0.682722E-03	7	7	4.8750
POLYMERI	0	0.115603E-03	2	2	194.8906
POOL	0	0.247573E-03	22	22	9.3906
PORE	0	0.810325E-03	5	5	25.6094
PRESSURE	0	0.852627E-02	24	18	336.0469
QUENCH1	0	0.785125E-02	6	6	0.2969
QUENCH2	0	0.743436E-03	7	6	0.1562
REA_DIF1	0	0.325531E-01	4	4	0.4219
REA_DIF2	0	0.940620E-02	11	11	0.6719
RESERVOI	0	0.163363E+00	21	12	62.9219
RICH_EQU	0	0.831304E+00	29	17	39.4062
RICH_PT	0	0.158267E-02	15	15	51.0469
RICH_XEN	0	0.130039E-03	7	7	3.5312
RIE_BND	0	0.109949E-02	8	7	27.1094
RIE_CD	0	0.465401E-03	4	4	39.0312
RIE_LAX	0	0.304039E-02	5	5	22.5469
RIE_LD	0	0.242371E-01	8	5	52.2500
RIE_SEWI	0	0.163370E-01	12	10	505.1250
RIE_SO	0	0.122775E-01	53	10	73.5625
RIE_SOD	0	0.169191E-01	53	12	115.1406
ROD	0	0.260926E-06	8	8	0.0781
SALINE	0	0.813993E-03	62	20	4.5156
SE_PULSE	0	0.340079E-01	12	11	28.9219
SETTLER	0	0.671660E-02	9	9	5.6406
SH_FRONT	0	0.295376E-07	26	24	0.8281
SHEAR	0	0.335067E+00	115	53	923.2812
SILICON	0	0.290348E-01	6	6	3.2656
SIN_GOR1	0	0.227460E-06	42	27	43.7812
SIN_GOR2	0	0.388294E-05	8	8	1.3750
SINGSTEP	0	0.143757E+01	51	30	206.5625
SKIN_X	0	0.429518E+00	200	75	19.2188
SKIN1	0	0.101111E-02	66	61	87.3906
SKIN10	0	0.101124E-02	71	66	80.6562
SKIN11	0	0.100319E+01	36	32	30.0625
SKIN12	0	0.411773E+07	29	29	59.9844
SKIN13	0	0.127420E-03	15	15	126.9844
SKIN14	0	0.136079E-03	13	13	46.0625
SKIN15	0	0.420707E+07	39	34	74.6250
SKIN16	0	0.659128E-15	9	7	5.0000
SKIN2	0	0.150624E-01	1215	100	26.4531
SKIN3	0	0.174374E-01	7	7	1.1094
SKIN3_X	0	0.992688E+02	45	41	49.5469
SKIN4	0	0.404347E-02	120	67	8.9375
SKIN5	0	0.128539E-01	66	62	56.6094
SLAB	0	0.198279E+00	59	33	83.4375
SLAB_CTR	0	0.155106E-05	40	40	18.2344
SOIL	0	0.158877E-01	27	24	10.9062
SOLID	0	0.116964E-03	9	9	0.2031
SOLITON	0	0.251399E-03	41	26	4.0156
SORP_IS1	0	0.208980E+00	8	8	0.2969
SORP_IS2	0	0.386411E-02	32	25	1.0469
SORP_IS3	0	0.833553E-03	5	5	20.5000
SORPTION	0	0.225283E-01	43	19	3.3906
SOUND	0	0.373050E-04	26	26	1.7031
SPHERE	0	0.282998E-02	4	4	0.1562
STAR_NET	0	0.724959E-04	8	8	0.5156
STARTBED	0	0.372554E-05	4	4	0.7812
STARTUP	0	0.320311E+00	6	6	13.0469

(continued)

name	i_f	$f(x^\star)$	n_{its}	n_f	*time*
STEPHAN	0	0.438109E-07	29	19	4.2812
STFFDET1	0	0.537536E-02	6	6	0.3594
STFFDET2	0	0.615633E-03	25	20	3.0312
STR_FISH	0	0.157315E-02	7	7	2.9375
T_DIFFUS	0	0.674100E-04	42	30	4.4062
TELEGRPH	0	0.154278E-02	27	22	4.4062
TIME_OPT	0	0.144639E-01	8	8	0.4375
TONGUE	0	0.129918E-01	17	14	0.6406
TRAFFIC	0	0.156873E+02	63	26	128.8750
TRAN_DEG	0	0.142085E-03	17	16	2.0469
TRANSDER	0	0.582492E-04	11	9	0.3906
TRANSMEM	0	0.198394E-03	139	86	10.3438
TRAV_WAV	0	0.400271E-03	43	33	13.7031
TUBE	0	0.877860E-01	4	4	7.5781
TUBE0	0	0.364738E-03	13	13	1.0000
TUNNEL	0	0.956775E-04	11	11	0.2500
TWO_POPS	0	0.471723E-02	14	14	0.6875
UNI_BEAM	0	0.345882E-02	10	10	155.3906
VIB_BEAM	0	0.279508E-03	23	8	1.6250
VISCOUS	0	0.125276E-08	12	12	9.6562
WATER	0	0.406317E-03	7	7	1.9375
WAVE1	0	0.223739E-09	27	18	0.9219
WAVE2	0	0.614900E-08	47	31	21.7500
WAVE3	0	0.433096E+01	6	6	0.3438
WAVE4	0	0.788310E-01	18	18	15.0312

APPENDIX B: The Modeling Language PCOMP

All nonlinear model functions defining fitting criteria, dynamical model equations and constraints, are implemented in a special modeling language called PCOMP, see Dobmann, Liepelt, Schittkowski and Trassl [76], Dobmann, Liepelt, Schittkowski [75], or Liepelt and Schittkowski [184].

The language format is similar to Fortran. The source code must first be pre-compiled. Proceeding from the intermediate code generated, function and gradient values are evaluated from the code during run time. A particular advantage is that gradients, as far as needed, are calculated automatically without any numerical approximation errors. Moreover, it is also possible to generate Fortran code for function and first and second derivatives.

The usage of the underlying codes is extremely flexible in the sense that special subsets of the given function can be selected, moreover special subsets of variables for which partial derivatives are to be computed.

5.3 Language Summary

The PCOMP-language is a subset of Fortran with a few extensions, see Dobmann, Liepelt, Schittkowski and Trassl [76] for details. In particular, the declaration and executable statements must satisfy the usual Fortran input format, i.e., must start at column 7. Comments beginning with `C` at the first column, may be included in a program text wherever needed. Statements may be continued on subsequent lines by including a continuation mark in the 6th column. Either capital or small letters are allowed for identifiers of the user and key words of the language. The length of an identifier has to be smaller than 20 tokens.

In contrast to Fortran, however, most variables are declared implicitly by their assignment statements. Variables and functions must be declared separately only if they are used for automatic differentiation. PCOMP possesses special constructs to identify program blocks.

* `PARAMETER`
  Declaration of constant integer parameters to be used throughout the program, particularly for dimensioning index sets.
* `SET OF INDICES`
  Definition of index sets that can be used to declare data, variables and functions or to define `sum` or `prod` statements.
* `INDEX`
  Definition of an index variable, which can be used in a `FUNCTION` program block.
* `REAL CONSTANT`
  Definition of real data, either without index or with one- or two-dimensional index. An index may be a variable or a constant number within an index set. Also arithmetic expressions may be included.
* `INTEGER CONSTANT`
  Definition of integer data, either without index or with one- or two-dimensional index. An index may be a variable or a constant number within an index set. Also arithmetic integer expressions may be included.
* `TABLE <identifier>`
  Assignment of constant real numbers to one- or two-dimensional array elements. In subsequent lines, one has to specify one or two indices followed by one real value per line in a free format (starting at column 7 or later).
* `VARIABLE`
  Declaration of variables with up to one index, with respect to which automatic differentiation is to be performed.

* `CONINT <identifier>`
  Declaration of a piecewise constant interpolation function.
* `LININT <identifier>`
  Declaration of a piecewise linear interpolation function.
* `SPLINE <identifier>`
  Declaration of a spline interpolation function.
* `MACRO <identifier>`
  Definition of a macro function, an arbitrary set of PCOMP statements that define an auxiliary function to be inserted into subsequent function declaration blocks. Macros are identified by a name that can be used in any right-hand side of an assignment statement.
* `FUNCTION <identifier>`
  Declaration of functions either with up to one index, for which function and derivative values are to be evaluated. The subsequent statements must assign a numerical value to the function identifier.
* `END`
  End of the program.

It is recommended to follow the order of the above program blocks. They may be repeated whenever desirable. Data must be defined before their usage in a subsequent block. All lines after the final `END` statement are ignored by PCOMP. The statements within the program blocks are very similar to usual Fortran notation and must satisfy the following guidelines:

Constant data: For defining real numbers either in analytical expressions or within the special constant data definition block, the usual Fortran convention can be used. In particular the `F-`, `E-` or `D-`format is allowed.

Identifier names: Names of identifiers for variables and functions, index sets and constant data, must begin with a letter and the number of characters, i.e. letters, digits and underscores, must not exceed 20.

Index sets: Index sets are required for the `SUM` and `PROD` expressions and for defining indexed data, variables and functions. They can be defined in different ways:

1. Range of indices,

```
ind1 = 1..27
```

2. Set of indices,

```
ind2 = 3,1,17,27,20
```

3. Computed index sets,

```
ind3 = 5*i + 100 , i=1..n
```

4. Parameterized index sets,

```
ind4 = n..m
```

Assignment statements: As in Fortran, assignment statements are used to assign a numerical value to an identifier, which may be either the name of the function that is to be defined, or of an auxiliary variable that is used in subsequent expressions,

```
r1  =  p1*p4 + p2*p4 + p3*p2 - 11
r2  =  p1 + 10*p2 - p3 + p4 + p2*p4*(p3 - p1)
f   =  r1**2 + r2**2
```

Analytical expressions: An analytical expression is, as in Fortran, any allowed combination of constant data, identifiers, elementary or intrinsic arithmetic operations and the special `SUM`– and `PROD`–statements. Elementary operations are

```
+ , - , * , / , **
```

Note that PCOMP handles integer expressions in exponents in the same way as real expressions, and one should avoid non-positive arguments. Integer constants are treated as usual artithmetic operations. Allowed intrinsic functions are

```
ABS, SIN, COS, TAN, ASIN, ACOS, ATAN, SINH, COSH
TANH, ASINH, ACOSH, ATANH, EXP, LOG, LOG10, SQRT
```

Alternatively, the corresponding double precision Fortran names possessing an initial `D` can be used as well. Brackets are allowed to combine groups of operations. Possible expressions are

```
5*DEXP(-z(i))
```

or

```
LOG(1 + SQRT(c1*f1)**2)
```

`INDEX`–**Variables:** In PCOMP it is possible to define indices separately to avoid unnecessary differentiation of integer variables. They have to be defined in the program block `INDEX`,

```
*     INDEX
      i,j
      l
```

It is allowed to manipulate the index by statements of the form

```
i = 1+2*4-3
i = a(1)
f = a(i+2)+i*2.0
f = SUM(a(m-i), m IN ind)
f = i
f = g(i)
```

In this case, `a` must be declared in form of in integer array. However, the following assignment statements are not allowed, if `b` is a real array:

```
i = b(3)
i = 1.0
i = 4/2
f(i) = 3.0
```

Interpolation functions: PCOMP admits the interpolation of user defined data, using either a piecewise constant, piecewise linear, or a cubic spline function. Given n pairs of real values (t_1, y_1), ..., (t_n, y_n), we are looking for a nonlinear function interpolating these data. In the first case, we define a piecewise constant interpolation by

$$c(t) = \begin{cases} 0 & , \ t < t_1 \ , \\ y_i & , \ t_i \le t < t_{i+1} \ , \ \ i = 1, \ldots, n-1 \ , \\ y_n & , \ t_n \le t \ . \end{cases}$$

A continuous piecewise liner interpolation function is

$$l(t) = \begin{cases} y_1 & , \ t < t_1 \ , \\ y_i + \dfrac{t - t_i}{t_{i+1} - t_i}(y_{i+1} - y_i) & , \ t_i \le t < t_{i+1} \ , \ \ i = 1, \ldots, n-1 \ , \\ y_n & , \ t_n \le t \ , \end{cases}$$

and a cubic spline is given by

$$s(t) = \begin{cases} p(t; t_1, t_2, t_3, t_4, y_1, y_2, y_3, y_4) & , \ t < t_4 \ , \\ \overline{s}(t; t_4, \ldots, t_n, y_4, \ldots, y_n, \dfrac{d}{dt} p(t_4; \ldots), 0) & , \ t_4 \le t \ , \end{cases}$$

where $p(t; t_1, t_2, t_3, t_4, y_1, y_2, y_3, y_4)$ is a cubic polynomial with

$$p(t_i; t_1, t_2, t_3, t_4, y_1, y_2, y_3, y_4) = y_i \ , \ \ i = 1, \ldots, 4 \ ,$$

and $\overline{s}(t; \overline{t_1}, \ldots, \overline{t}_m, \overline{y}_1, \ldots, \overline{y}_m, \overline{y}'_1, \overline{y}'_m)$ a cubic spline function interpolating $(\overline{t}_1, \overline{y}_1)$, ..., $(\overline{t}_m, \overline{y}_m)$ subject to the boundary conditions

$$\frac{d}{dt}\overline{s}(\overline{t}_i; \overline{t_1}, \ldots, \overline{t}_m, \overline{y}_1, \ldots, \overline{y}_m, \overline{y}'_1, \overline{y}'_m) = \overline{y}'_i \ , \ \ i = 1 \text{ and } i = m \ .$$

It is essential to understand that the constant and spline interpolation functions are not symmetric. Our main interest are dynamical systems, say ordinary or partial differential equations, where the initial value is set to 0 without loss of generality, leading to a non-symmetric domain. Moreover, interpolated data are often based on experiments that reach a steady state, i.e., a constant value. Thus, a zero derivative is chosen at the right end point for spline interpolation to facilitate the input of interpolated steady state data. On the other hand, any other boundary conditions can be enforced by adding artificial interpolation data.

The spline functions generated, are twice differentiable with the exception of the fourth break point. At this point, there exists only the first derivative and PCOMP generates the right-hand side differential quotient for the second derivative. We need at least four pairs of data to construct a spline interpolation as outlined above.

To given an example, we assume that we want to interpolate the nonlinear function $f(t)$ given by the discrete values $f(t_i) = y_i$ from Table 1, using the different techniques mentioned above.

Interpolation functions are defined by a program block starting with the keyword `CONINT` for piecewise constant functions, `LININT` for piecewise linear functions, or `SPLINE` for piecewise cubic splines, followed by the name of the function. The numerical values of the break points and the function values are given on the subsequent lines, using any standard format starting at column 7 or later. Using piecewise constant approximations, we get for our example:

```
*     CONINT F
      0.0  0.00
```

i	t_i	y_i
1	0.0	0.00
2	1.0	4.91
3	2.0	4.43
4	3.0	3.57
5	4.0	2.80
6	5.0	2.19
7	6.0	1.73
8	7.0	1.39
9	8.0	1.16
10	9.0	1.04
11	10.0	1.00

Table 1: Interpolation data

```
1.0  4.91
2.0  4.43
3.0  3.57
4.0  2.80
5.0  2.19
6.0  1.73
7.0  1.39
8.0  1.16
9.0  1.04
10.0 1.00
```

Within a function definition block, the interpolation functions are treated as intrinsic Fortran functions, that is, they have to contain a variable or constant as a parameter. If we assume that `T` has previously been declared as a variable, a valid statement could look like

```
*     FUNCTION OBJ
      OBJ = F(T)
```

Macros: PCOMP does not allow the declaration of subroutines. However, it is possible to define macros, that is, arbitrary sequences of PCOMP statements that define an auxiliary variable to be inserted into the beginning of subsequent function declaration blocks. Macros are identified by a name that can be used in any right-hand side of an assignment statement

`*` `MACRO` ⟨identifier⟩

followed by a group of PCOMP statements that assign a numerical value to the given identifier. This group of statements is inserted into the source code block that contains the macro name. Macros have no arguments, but they may access all variables, constants, or functions that have been declared up to their first usage. Any values assigned to local variables within a macro, are also available outside in the corresponding function block.

If we assume that `x` is a variable and we want to define a macro that computes the square of `x`, we would write something like

```
*     MACRO sqr
      sqr = x*x
```

Now it is possible to replace each occurrence of the term `x*x` with an invocation of the macro that we have just defined, for example

```
f = sqr-5.2
```

`SUM`– **and** `PROD`–**expressions:** Sums and products over predetermined index sets are formulated by `SUM` and `PROD` expressions, where the corresponding index and the index set must be specified, for example in the form

```
f = 100*PROD(p(i)**a(i), i IN inda)
```

In the above example, `p(i)` might be a variable vector defined by an index set, and `a(i)` an array of constant data.

Control statements: To control the execution of a program, the conditional statements

```
IF ⟨condition⟩ THEN
   ⟨statements⟩
ENDIF
```

or

```
IF ⟨condition⟩ THEN
   ⟨statements⟩
ELSE
   ⟨statements⟩
ENDIF
```

can be inserted into a program. Conditions are defined as in Fortran by the comparative operators `.EQ.`, `.NE.`, `.LE.`, `.LT.`, `.GE.`, `.GT.`, which can be combined using brackets and the logical operators `.AND.`, `.OR.` and `.NOT.`.
The `GOTO`– and the `CONTINUE`–statements are further possibilities to control the execution of a program. The syntax for these statements is

```
GOTO ⟨label⟩
```

and

```
⟨label⟩ CONTINUE
```

where `label` has to be a number between 0 and 9999. Since PCOMP produces labels during the generation of the Fortran code in the reverse mode, it is advisable to use labels between 5000 and 9999. The `<label>` part of the `CONTINUE`-statement has to be located between columns 2 and 5 of an input line. Together with an index, the `GOTO`-statement can be used for example to simulate `DO`-loops, which are forbidden in PCOMP,

```
      i = 1
      s = 0.0
 6000 CONTINUE
      s = s + a(i)*b(i)
      i = i+1
      IF (i.LE.n) THEN
        GOTO 6000
      ENDIF
```

Whenever indices are used within arithmetic expressions, it is allowed to insert polynomial expressions of indices from a given set. However, functions must be treated in a particular way. Since the design goal is to generate short, efficient Fortran codes, indexed function names can be used only in exactly the same way as defined. In other words, if a set of functions is declared by

```
* FUNCTION  f(i),  i IN index
```

then only an access to `f(i)` is allowed, not to `f(1)` or `f(j)`, for example. In other words, PCOMP does not enroll indexed functions to a sequence of single expressions similar to the treatment of `SUM` and `PROD` statements.

In PCOMP, it is allowed to pass variable values from one function block to the other. However, the user must be aware of a possible failure, if in the calling program the evaluation of a gradient value in the first block is skipped.

One should be very careful when using the conditional statement `IF`. Possible traps that prevent a correct differentiation are reported in Fischer [100], and are to be illustrated by an example. Consider the function $f(p) = p^2$ for $n = 1$. A syntactically correct formulation would be:

```
IF (p.EQ.1) THEN
  f = 1
ELSE
  f = p**2
ENDIF
```

In this case, PCOMP would try to differentiate both branches of the conditional statement. If p is equal to 1, the derivative value of f is 0; otherwise it is $2p$. Obviously we get a wrong answer for $p = 1$. This is a basic drawback for all automatic differentiation algorithms of the type under consideration.

PCOMP allows the execution of external statements that must be linked to PCOMP in a special way, see Dobmann, Liepelt, Schittkowski and Trassl [76].

A frequently needed computational value in case of a PDE model is the integral with respect to the spatial variable x, i.e.,

$$\int_{x_{j-1}^a}^{x_j^a} u^i(p, x, t)dx$$

where the integral is taken over the j-th area where the PDE is defined, $j = 1, \ldots, n_t$. Index i denotes the i-th solution component we want to integrate, $i = 1, \ldots, n_p$. The integral is evaluated by Simpson's rule and denoted by

```
SIMPSN(I,J)
```

in the PCOMP language. This name can be inserted in an arithmetic expression, for example to compute a fitting criterion. The corresponding time value is either a measurement value or an intermediate value needed for generating plot data.

5.4 Error Messages

PCOMP reports error messages in the form of integer values of the variable IERR and, whenever possible, also line numbers LNUM. The meaning of the messages is listed in the following table. Note that the corresponding text is displayed if the error routine SYMERR is called with parameters LNUM and IERR.

Syntax errors are reported when starting the parser, or during run time. The corresponding error code and a line number are passed to the calling program.

1 - file not found
2 - file too long
3 - identifier expected
4 - multiple definition of identifier
5 - comma expected
6 - left bracket expected
7 - identifier not declared
8 - data types do not fit together
9 - division by zero
10 - constant expected
11 - operator expected
12 - unexpected end of file
13 - range operator '..' expected
14 - right bracket ')' expected
15 - 'THEN' expected
16 - 'ELSE' expected
17 - 'ENDIF' expected
18 - 'THEN' without corresponding 'IF'
19 - 'ELSE' without corresponding 'IF'
20 - 'ENDIF' without corresponding 'IF'
21 - assignment operator '=' expected
22 - wrong format for integer number
23 - wrong format for real number
24 - formula too complicated
25 - error in arithmetic expression
26 - internal compiler error
27 - identifier not valid
28 - unknown type identifier
29 - wrong input sign
30 - stack overflow of parser
31 - syntax error
32 - available memory exceeded
33 - index or index set not allowed
34 - error during dynamic storage allocation
35 - wrong number of indices
36 - wrong number of arguments
43 - number of variables different from declaration
44 - number of functions different from declaration
45 - END - sign not allowed
46 - Fortran code exceeds line

47 - **: domain error
48 - bad input format
49 - length of working array IWA too small
50 - length of working array WA too small
51 - ATANH: domain error
52 - LOG: domain error
53 - SQRT: domain error
54 - ASIN: domain error
55 - ACOS: domain error
56 - ACOSH: domain error
57 - LABEL defined more than once
58 - LABEL not found
59 - wrong index expression
60 - wrong call of the subroutine SYMINP
61 - wrong call of the subroutine SYMPRP
62 - compilation of the source file in GRAD-mode
63 - interpolation values not in right order
64 - not enough space for interpolation functions in subroutine REVCDE
65 - length of working array IWA in subroutine SYMFOR too small
66 - not enough interpolation values
67 - compilation of source file not in GRAD-mode
68 - missing macro name
69 - too many macros defined