

# A continuous genetic algorithm for the calibration of a sedimentation model

S. Berres <sup>a</sup>, A. Coronel <sup>b</sup> and R. Lagos <sup>c</sup>

<sup>a</sup>*Departamento de Ciencias Matemáticas y Físicas, Facultad de Ingeniería,  
Universidad Católica de Temuco, Chile*

<sup>b</sup>*GMA, Departamento de Ciencias Básicas, Facultad de Ciencias, Universidad del Bío-Bío,  
Campus Fernando May, Chillán, Chile*

<sup>c</sup>*Departamento de Matemática y Física, Facultad de Ciencias, Universidad de Magallanes,  
Punta Arenas, Chile  
Email: [sberres@uct.cl](mailto:sberres@uct.cl)*

**Abstract:** In this contribution we consider the problem of flux identification in a scalar conservation law modeling the phenomenon of sedimentation. The experimental observation data used for the calibration consist of a solid concentration profile at a fixed time. The identification problem is formulated as an optimization one, where the distance between the profiles of the model simulation and observation data is minimized by a least squares cost function.

The direct problem is approximated by a monotone finite volume scheme. The numerical solution of the calibration problem is obtained by a continuous genetic algorithm. Numerical results are presented in order to validate the efficiency of the proposed algorithm. The optimization by a continuous genetic algorithm turns out to be more robust than previous calibration attempts with a gradient method.

The calibration of the model uses a set of measurements along the column at a fixed time  $t = T$ . The  $n$  measurement locations are denoted as  $\bar{x}_i$ . The solids concentration values measured in these points are  $\bar{\phi}_i$  which means that the data set for the calibration is specified as

$$\left\{ (\bar{x}_i, \bar{\phi}_i) \in [0, H] \times [0, u_{\max}] : 0 = \bar{x}_0 < \bar{x}_1 < \dots < \bar{x}_{n-1} < \bar{x}_n = H \right\}.$$

In practice, the data are represented by a curve, i.e. one defines  $u^{\text{exp}} : [0, H] \rightarrow [0, u_{\max}]$  such that  $u^{\text{exp}}(\bar{x}_i) = \bar{\phi}_i$ . In terms of this experimental information, the inverse problem of determining the flux function is formulated in an abstract setting as the optimization problem

$$\text{minimize} \quad J(u, f) := \frac{1}{2} \sum_{i=0}^n \left| (u - u^{\text{exp}})(\bar{x}_i) \right|^2, \quad (1)$$

where  $u$  is the solution of the direct problem for the given parameter set, and  $f = f(u)$  is the flux function of the governing equation

$$u_t + f(u)_x = 0, \quad \text{for } x \in (0, H), \quad f(u) = 0 \quad \text{for } x \in \{0, H\}. \quad (2)$$

To handle the problem (1) it is reduced to a parameter identification problem, where one assumes that the flux function  $f$  is parametrized having a parameter vector  $\mathbf{e} = (e_1, \dots, e_d) \in \mathbb{R}^d$ , which means  $f(\cdot) = f(\cdot; \mathbf{e})$ . In consequence, the parameter identification is formulated by the following optimization problem in several variables:

$$\text{minimize} \quad \mathcal{J}(\mathbf{e}) := J(u(\mathbf{e}), f(u(\mathbf{e}); \mathbf{e})) \quad (3)$$

The search of the minimum of the cost function (3) follows the first-discretize-then-optimize paradigm, using a standard finite-volume scheme for solving the direct problem.

**Keywords:** *Continuous genetic algorithm, calibration, parameter identification, sedimentation*

## 1 INTRODUCTION

Sedimentation is a mechanical process used for separating the components of a solid-liquid mixture. Its remarkable benefits in industrial processing promoted it to a relevant phenomenon for scientific research, see Bustos et al. (1999). The guidelines of sedimentation theory for incompressible materials were initially established by Kynch (1952). This theory proposes conditions under which sedimentation processes can be modeled using a nonlinear one-dimensional transport equation in form of a scalar conservation law. Although these assumptions are idealizing, they give a very reasonable explanation of the phenomenon. Up to now, several extensions of this theory have been developed; the main historical aspects of the development of the corresponding theory are detailed in Bustos et al. (1999); Bürger and Wendland (2001); Concha and Bürger (2003).

The key component of a conservation law is the flux function, which describes the properties of the used material in dependence of the solids concentration as unknown variable. In the model calibration problem one starts with a solids concentration profile at a fixed time and supposes a parametric form of the flux function. The parametrization of the flux function is a result of empirically verified constitutive equations. The calibration problem is formulated as an optimization problem, where a parameter vector is searched that minimizes the distance between the solution profile of the model and the observed profile.

In the recent decades, the calibration problem has received considerable attention, see Berres et al. (2005); Bürger et al. (2008, 2009), where the optimization problem was solved by a gradient method. However, the rigorosity of this approach is unresolved, because the entropy solution of a scalar conservation law is typically a discontinuous function, even when the initial condition and the flux function has a high regularity. The low regularity of the solution of the state equations implies a low regularity of the cost function. In particular, the differentiability of the cost function is a complicated issue.

In this contribution, the optimization problem is solved by applying a continuous genetic algorithm. The cost function is discretized using the numerical solution of the direct problem. The natural numerical approximation of the direct problem is made using the finite volume method, where the numerical flux is according to Engquist and Osher (1981). The numerically constructed cost function defines an integer programming problem. Therefore, the application of a binary coded genetic algorithm is very slow such that we opt for the implementation of a continuously encoded genetic algorithm, for details consult the book of Haupt and Haupt (2004) and also the references Michalewicz (1992); Holland (1975); Goldberg (1989). The continuous genetic algorithm proposed for the calibration is applied to synthetic observation data. This contribution is a draft version of a more extended paper, where the performance of the algorithm is tested by several examples.

The present contribution is organized in four sections. In Section 2 the modeling hypothesis of the Kynch model are stated and the sedimentation model is deduced and genetic algorithm is outlined. In Section 3 the numerical results of the parameter identification are presented. Finally, in Section 4 some conclusions of the work are stated.

## 2 THEORY

### 2.1 Mass and momentum balance of a mixture

The theoretical basis of mixtures within continuum mechanics is established in the works of Fick, Stefan and Maxwell, the formalization in the theoretical framework known today was established by Truesdell. This theory assumes that every point in the space can be occupied by a finite number of different particles, one for each system component. Consequently, the mixture can be represented as the superposition of several continuous media, each of which maintains its own movement together with the specifications imposed by the interactions. In the theory of mixtures each component satisfies a balance equation. If an extensive property per mass unit of the  $i$ -th component of the mixture with density  $\rho_i$  is denoted by  $G_i$  then

$$\frac{\partial}{\partial t}(\rho_i G_i) + \nabla \cdot J_i - r_i = 0, \quad (4)$$

where  $J_i$  is the density flux and  $r_i$  is the generation rate of the  $i$ -th component per volume unit. The flux can be separated into its convective and diffusive part and establish the following equation

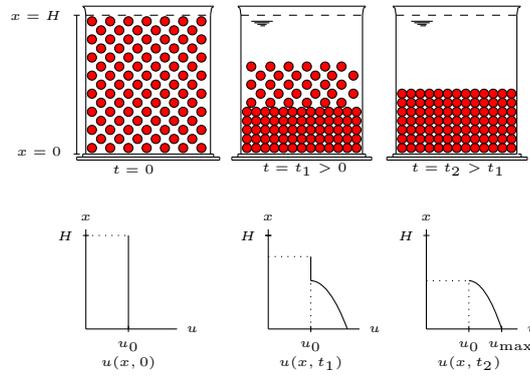
$$\frac{\partial}{\partial t}(\rho_i G_i) + \nabla \cdot (\rho_i G_i \mathbf{v}) + \nabla \cdot J_D - r_i = 0, \quad (5)$$

where  $\mathbf{v}$  is the mean velocity of the mixture and  $J_D$  is the diffusive flux. The balance equation (5) can be used to derive application specific mass and momentum balances.

## 2.2 Mathematical sedimentation model

The mathematical sedimentation model describes a mixture of solid particles immersed in a fluid, see Figure 1 for a schematic illustration. It is assumed that the mixture satisfies the following properties:

- (A1) All solid particles are of the same size, shape and density.
- (A2) The solid and fluid of the mixture are incompressible. There is no mass transfer between the components.
- (A3) The relative solid-liquid velocity is  $\mathbf{v}_r = \mathbf{v}_s - \mathbf{v}_f$ , where  $\mathbf{v}_s$  is the velocity of the solid and  $\mathbf{v}_f$  is the velocity of the liquid, dependent only on the local solids concentration  $u$ .
- (A4) Wall effects are negligible.



**Figure 1.** Separation process within sedimentation column

The mass balances for the solid and fluid phase are

$$u_t + \nabla \cdot (u\mathbf{v}_s) = 0, \quad (1-u)_t + \nabla \cdot ((1-u)\mathbf{v}_f) = 0, \quad (6)$$

respectively. Defining the mean volumetric velocity  $\mathbf{q}(\mathbf{x}, t) = u\mathbf{v}_s + (1-u)\mathbf{v}_f$  and summing up both equations one obtains  $\nabla \cdot (\mathbf{q}(\mathbf{x}, t)) = 0$ . From the mass balance equations for the solids and fluid phase it can be deduced that for the sedimentation bath the differential equation modeling the problem has the form

$$u_t + (f_b(u))_x = 0, \quad x \in [0, H], \quad t > 0, \quad (7)$$

where  $H$  denotes the height of the mixture contained in the column. According to the empirical evidences, the properties which the flux functions needs to satisfy are

$$f_b(u) < 0, \quad u \in (0, u_{\max}), \quad f_b(0) = f_b(u_{\max}) = 0, \quad f'_b(0) < 0 \quad \text{and} \quad f'_b(u_{\max}) > 0. \quad (8)$$

In practice, this constraint is implicitly imposed by the choice of the parametric form of the flux function. In the example, the flux function

$$f_b(u) = v_\infty u \left(1 - \frac{u}{u_{\max}}\right)^C, \quad (v_\infty, C, u_{\max}) \in (-\infty, 0) \times (1, \infty) \times (0, 1]. \quad (9)$$

is considered. Note that the negative  $v_\infty < 0$  complies with the constraint  $f_b(u) < 0$ . The parameters which are identified are  $v_\infty$ ,  $C$  and  $u_{\max}$ , establishing the parameter vector

$$\mathbf{e} = (e_1, e_2, e_3) = (v_\infty, C, u_{\max}) \quad (10)$$

with  $d = 3$  elements.

The equation (7) is a non-linear first order hyperbolic equation and should be supplemented by adequate initial and boundary conditions. The conditions (K3) and (K4) imply

$$u(x, 0) = u_0, \quad x \in [0, H], \quad u(0, t) = u_{\max}, \quad t > 0, \quad u(H, t) = 0, \quad t > 0, \quad (11)$$

where  $u_0$  and  $u_{\max}$  are constants such that  $0 < u_0 < u_{\max} < 1$ . In consequence, the sedimentation model in the column is given by the equation (7) with initial and boundary conditions given by (11) and a flux function satisfying (8). Equivalently to the boundary conditions stated in (11) one can set zero-flux boundary conditions. The reason is that during a downwards moving flow a zero-concentration at the top and a full packing at the bottom have the same effect as walls.

### 2.3 Genetic algorithm

Evolutionary computation techniques applied to numerical optimization mimic the principles of natural selection formulated by Darwin. The foundations of evolutionary computation are the following four paradigms: Genetic algorithms (Holland, 1975), genetic programming (Koza, 1992), evolutionary strategies (Rechenberg, 1973) and evolutionary programming (Fogel et al., 1966). Genetic algorithms are the most popular technique because of their simplicity of implementation, global convergence and several other advantages extensively described in Sivanandam and Deepa (2008). The first genetic algorithm was proposed by Holland (1975) in his pioneering work. Following this work, various improvements have been developed and proposed by several researchers. The most complete and technically advanced reference is the book of Michalewicz (1992). In the following the standard terminology of genetic algorithms is used. For completeness of the presentation the main concepts are recalled: chromosomes, genes, population and generation, for more details see Haupt and Haupt (2004); Holland (1975); Sivanandam and Deepa (2008).

The definitions of basic concepts are as follows:

**Definition 1.** *A chromosome is an array of parameters that parametrizes the cost function and is subject to be identified.*

**Definition 2.** *A gene corresponds to one of the components of the parameter vector; thus a gene is part of the chromosome.*

**Definition 3.** *A population is the set of chromosomes.*

**Definition 4.** *A generation is the population defined after each iteration of the genetic algorithm.*

Usually chromosomes are encoded at the birth of each generation. The mostly used genetic algorithms select a representation either as a binary number or as a floating-point. Taking into account the requirements of the problem, in this contribution the second option is chosen, since the representation of the physical parameters require many bits. In fact, in the considered situation, a real coded genetic algorithm is the natural choice, since the variable of the parameter space is continuous. On the other hand, for general optimization issues, this formal choice is less relevant in view of the fact that the genetic operations in the implemented algorithms are decisive for its performance (Gaffney et al., 2010). If the cost function is continuous then the chromosomes do not require to be encoded before being evaluated by the cost function. See the book of Haupt and Haupt (2004) on the advantages and disadvantages of both representations.

The continuous genetic algorithm consists of two major steps, the selection of the initial population and the natural selection. More specifically it has the following description:

(a) **Initial population.** Definition of a matrix that stores the initial random population:

$$\mathbb{E} = \left[ \mathbf{e}_1 | \mathbf{e}_2 | \cdots | \mathbf{e}_n \right]^t,$$

where each row  $\mathbf{e}_j \in \Omega \subset \mathbb{R}^d$ ,  $j = 1, \dots, n$ , belongs to the hypercube  $\Omega$  defined as  $\Omega := \prod_{i=1}^d [l_i, u_i] \subset D$ , with  $l_i < u_i$  for  $i = 1, \dots, d$ .

(b) **Natural selection.**

- (b1) **Cost of the population.** Definition of the vector  $\mathbf{cost} := (\mathcal{J}_\Delta(\mathbf{e}_1), \dots, \mathcal{J}_\Delta(\mathbf{e}_n)) \in \mathbb{R}^n$  by the evaluation of the cost function in each of the chromosomes of the population  $\mathbb{E}$ . Definition of the aggregated matrix  $\hat{\mathbb{E}} = [\mathbb{E}|\mathbf{cost}]$ . If there exists some  $\ell \in \{1, \dots, n\}$  such that  $\mathcal{J}_\Delta(\mathbf{e}_\ell) \leq J_{\min}$ , then the solution of discrete minimization problem is sufficiently approximated by  $\mathbf{e}_\ell$  and the iteration is stopped. The notation  $J_{\min}$  is used for the tolerance of the evaluation of the cost function.
- (b2) **Selection of the parents.** The parents are selected in three steps. Firstly, the matrix  $\hat{\mathbb{E}}$  is updated permuting the columns such that  $\hat{\mathbb{E}}_{1,d+1} \leq \hat{\mathbb{E}}_{2,d+1} \leq \dots \leq \hat{\mathbb{E}}_{n,d+1}$ . Secondly, if  $s \in (0, 1]$  denotes the selection rate, then the first  $\llbracket ns \rrbracket$  rows of  $\hat{\mathbb{E}}$  are selected and the sub-matrix of the so-called mating pool matrix  $\mathbb{F}$  is stored. The notation  $\llbracket \cdot \rrbracket$  is used for the upper integer function. Thirdly, applying a stochastic process like the rule of roulette, the parents are selected from the chromosomes of  $\mathbb{F}$ , see Haupt and Haupt (2004) for technical details.
- (b3) **Mating.** For the interbreeding of the parents an algebraic rule is defined. In this contribution the new chromosomes are obtained by applying a convex combination of the parents genes. The mating process ends when  $n - \llbracket ns \rrbracket$  new chromosomes are generated. The matrix  $\mathbb{E}$  is updated considering that the parents are stored in the first  $\llbracket ns \rrbracket$  rows and the children in the following rows.
- (b4) **Mutation.** Let  $\mu \in [0, 1]$  denote the mutation rate and  $m := \llbracket \mu(n-1)d \rrbracket$  the total number of mutations. During the mutation process, repeatedly  $m$  times, the random gene  $\mathbb{E}_{i,j}$  is replaced by a random number equally distributed in the range  $[l_j, u_j]$ . One observes that after the mutation the matrix  $\mathbb{E}$  is naturally updated.

For more details on the genetic algorithm see Haupt and Haupt (2004); Sivanandam and Deepa (2008).

### 3 RESULTS OF THE IDENTIFICATION

In this Section the continuous genetic algorithm is applied to the identification of parameters of the flux function  $f_b$ . For the genetic algorithm the parameter configuration suggested by Haupt and Haupt (2004) is used:

Parameter name	Value
Population ( $n$ )	20
Selection rate ( $s$ )	0.5
Cost tolerance ( $J_{\min}$ )	$1.0 \times 10^{-6}$
Mutation rate ( $\mu$ )	0.37
Maximum of iterations	20

It is assumed that the sensors are located in  $\bar{x}_i = \ell/20, i = 0, \dots, 20$ . In the example we consider a flux function defined by (9) where the parameters  $v_\infty = -2.7 \times 10^{-4}$  and  $u_{\max} = 0.5$  are prescribed such that  $C$  remains as only free parameter. In order to obtain synthetic experimental data a simulation of the direct problem is performed with observation time  $T = 3500$ , initial condition  $u_0(x) = 0.05$ , mesh refinement in terms of number of volumes  $M = 800$  and target parameter  $C = 5$ .

The set of values for the synthetic experimental data obtained by simulation of the direct problem are

$$\bar{\phi}_0 = 0.49759914, \quad \bar{\phi}_1 = 0.31229822, \quad \bar{\phi}_2 = 0.25193333, \quad \bar{\phi}_i = 0 \quad \text{for } i = 3, \dots, 20. \quad (12)$$

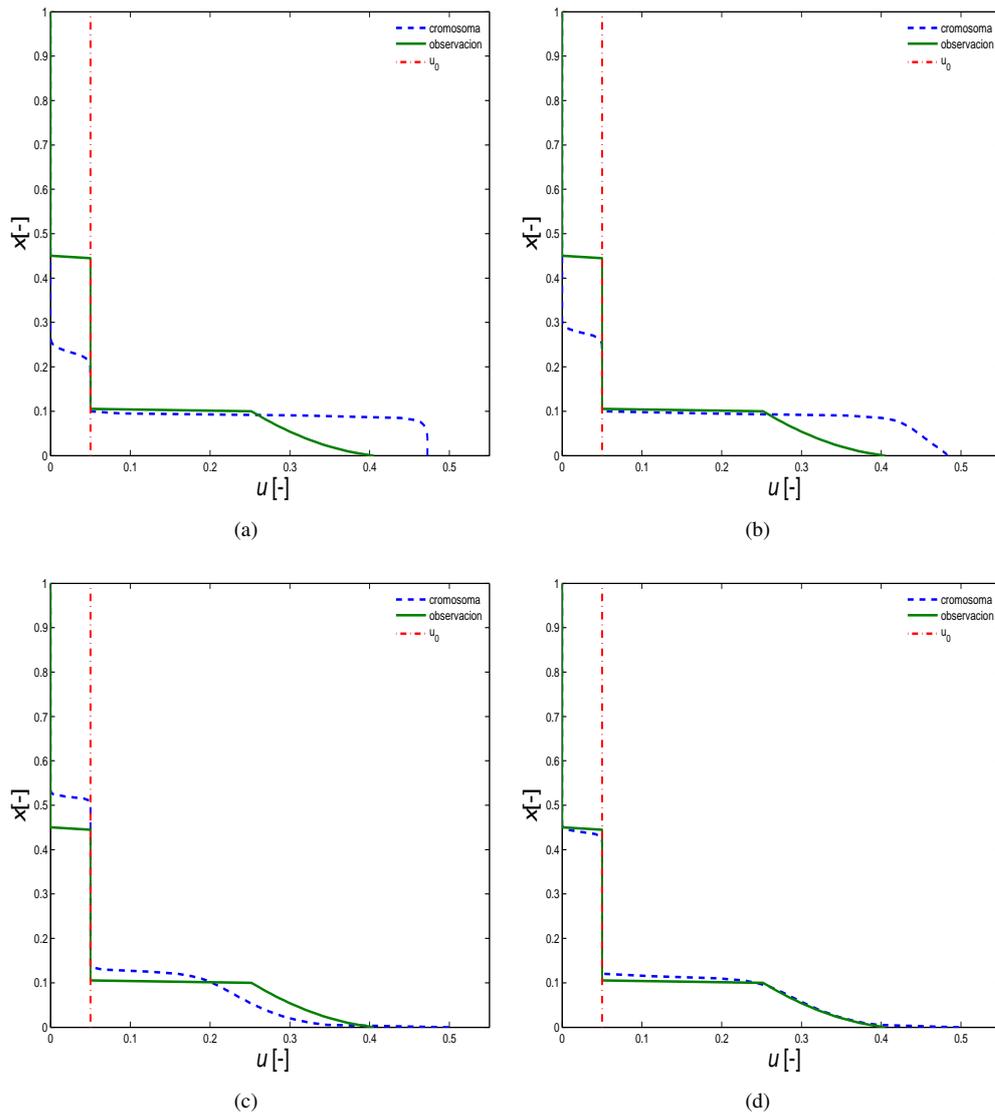
Thus one obtains the following fitted function as observation profile for  $T = 3500$ :

$$u^{\text{exp}}(x) = \begin{cases} (b^2 - \sqrt{b^2 - 4a(c - x/T)})/2a, & x \in [0, 0.1] \\ 0 & \text{otherwise,} \end{cases} \quad (13)$$

with  $a = 8.29285e-4, b = 7.52058e-4$  and  $c = 1.69437$ . The values  $a, b, c$  are chosen such that the function (13) fits best to the data (12). For the genetic algorithm we set  $\Omega = [2, 7]$  as search range for the parameter to identify. See Figure 2 for the comparison of the observed profile with the profiles obtained through the random chromosomes.

### 4 CONCLUSIONS

In this contribution a continuous genetic algorithm has been successfully validated in order to resolve the calibration problem of a flux function in a conservation law modeling the physical phenomenon of sedimentation of solid particles immersed in a fluid, where the flux function satisfies physical considerations introduced



**Figure 2.** Example 1: (a)–(c) Random initial population of some selected chromosomes, (d) Profile of the best chromosome at the 20-th generation.

by Kynch (1952). The distance between observed and identified profiles permits to conclude that the genetic algorithm resolves adequately the parameter identification problem for scalar non-convex conservation laws.

An unexpected observation with respect to the performance of the genetic algorithm is that in some test cases (see full paper version for details) the cost function of the best chromosome remains at the same level for various generations, and then suddenly decreasing, in order to remain again several steps in the next lower level. This means that, whereas the overall chromosome pool is permanently improving, the best chromosome usually maintains its composition during a certain period.

It is also noted that this new algorithm exceeds disadvantages of previously studied calibration methods like the gradient based optimization in Berres et al. (2005); Bürger et al. (2008, 2009), which include the following aspects: (a) the genetic algorithm randomly chooses the starting point for the iteration and (b) the genetic algorithm does not need an assumption on the differentiability of the cost function.

The identification algorithm turned out to be stable in spite of perturbed synthetic observation data (see full version for documentation of examples). The considered inverse problem is overdetermined, since, in the considered example as well as in practically relevant situations there are much more observation data (here 20, in industrial applications up to 1000) than free parameters to fit (usually never larger than 5). This overdeter-

mination classifies the considered inverse problem as a regression problem, which infact demonstrated to be stable with respect to experimental errors, as can be judged from the stable identifications even though the data are perturbed by noise.

#### ACKNOWLEDGEMENT

SB thanks the support of Conicyt (Chile) through Fondecyt project #1120587. AC thanks for the support of Fondecyt project # 11060400; and the research projects 124109 3/R, 104709 01 F/E and 121909 GI/C at Universidad del Bío-Bío, Chile.

#### REFERENCES

- Berres, S., B., A. R., Coronel, and M. Sepúlveda (2005). Numerical identification of parameters for a strongly degenerate convection-diffusion problem modelling centrifugation of flocculated suspensions. *Appl. Numer. Math.* 52(4), 311–337.
- Bürger, R., A. Coronel, and M. Sepúlveda (2008). *A numerical descent method for an inverse problem of a scalar conservation law modelling sedimentation*, pp. 225–232. Numerical Mathematics and Advanced Applications: Numerical Mathematics and Advanced Applications: Proceedings ENUMATH 2007, Springer Verlag, Providence, RI.
- Bürger, R., A. Coronel, and M. Sepúlveda (2009). *Numerical solution of an inverse problem for a scalar conservation law modelling sedimentation*, pp. 445–454. Hyperbolic problems: theory, numerics and applications, volume 67 of *Proc. Sympos. Appl. Math.*, pages . Amer. Math. Soc., Providence, RI.
- Bürger, R. and W. L. Wendland (2001). Sedimentation and suspension flows: historical perspective and some recent developments. *J. Engrg. Math.* 41, 101–116.
- Bustos, M. C., F. Concha, R. Bürger, and E. M. Tory. (1999). *Sedimentation and thickening*, Volume 8. Mathematical Modelling: Theory and Applications, Kluwer Academic Publishers, Dordrecht.
- Concha, F. and R. Bürger (2003). Thickening in the 20th century: a historical perspective. *Minerals & Metallurgical Process* 20 (2), 57–67.
- Engquist, B. and S. Osher (1981). One-sided difference approximations for nonlinear conservation laws. *Math. Comput.* 36, 321–351.
- Fogel, L. J., A. J. Owens, and M. J. Walsh. (1966). *Artificial intelligence through simulated evolution*. Wiley, Chichester, WS, UK.
- Gaffney, J., C. Pearce, and D. Green. (2010). Binary versus real coding for genetic algorithms: A false dichotomy? *ANZIAM J.* 51, C347–C359.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning* (1 ed.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Haupt, R. L. and S. E. Haupt (2004). *Practical genetic algorithms*. Wiley-Interscience [John Wiley & Sons].
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, USA.
- Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.
- Kynch, G. J. (1952). A theory of sedimentation. *Trans. Faraday Soc* 48, 166–176.
- Michalewicz, Z. (1992). *Genetic algorithms + data structures = evolution programs*. Artificial Intelligence. Springer-Verlag, Berlin.
- Rechenberg, I. (1973). *Evolutionstrategie: Optimierung Technischer Systeme nach Prinzipien des Biologischen Evolution*. Fromman-Holzboog Verlag, Stuttgart.
- Sivanandam, S. N. and Deepa (2008). *Introduction to genetic algorithms*. Springer, Berlin.