ASC Report No. 43/2011

Numerical Solution of the Density Profile Equation with p-Laplacians

G. Hastermann, P. Lima, L. Morgado, E. Weinmüller

Institute for Analysis and Scientific Computing
Vienna University of Technology — TU Wien
www.asc.tuwien.ac.at ISBN 978-3-902627-04-9

Most recent ASC Reports

42/2011 *M. Aurada, J.M. Melenk, D. Praetorius*
Mixed Conforming elements for the large-body limit in micromagnetics: a finite element approach

41/2011 *P. Amodio, T. Levitina, G. Settanni, E.B. Weinmüller*
On the Calculation of the Finite Hankel Transform Eigenfunctions

40/2011 *D.P. Hewett, S. Langdon, J.M. Melenk*
A high frequency hp boundary element method for scattering by convex polygons

39/2011 *A. Jüngel*
Semiconductor Device Problems

38/2011 *L. Neumann, A. Arnold, W. Hochhauser*
Zur Stabilität von geklebten und geklotzten Glasscheiben: Beurteilung der Dunkerley'schen Geraden zur Beulwertbestimmung

37/2011 *A. Dick, O. Koch, R. März, E. Weinmüller*
Collocation Schemes for Nonlinear Index 1 DAEs with a Singular Point

36/2011 *J.A. Carrillo, S. Hittmeir, A. Jüngel*
Cross diffusion and nonlinear diffusion preventing blow up in the Keller-Segel model

35/2011 *P. Fuchs, A. Jüngel, M. von Renesse*
On the Lagrangian structure of quantum fluid models

34/2011 *W. Auzinger, O. Koch, M. Thalhammer*
Defect-based local error estimators for splitting methods, with application to Schrödinger equations Part I. The linear case

33/2011 *A. Jüngel, R. Weishäupl*
Blow-up in two-component nonlinear Schrödinger systems with an external driven field

Institute for Analysis and Scientific Computing
Vienna University of Technology
Wiedner Hauptstraße 8–10
1040 Wien, Austria

E-Mail: admin@asc.tuwien.ac.at
WWW: http://www.asc.tuwien.ac.at
FAX: +43-1-58801-10196

ISBN 978-3-902627-04-9

Numerical Solution of the Density Profile Equation with p-Laplacians

G. Hastermann, *Institute for Analysis and Scientific Computing, Vienna University of Technology, Austria*
P.M. Lima, *CEMAT, Instituto Superior Técnico, UTL, Lisboa, Portugal*
M.L. Morgado, *CM-UTAD and Dept. Matemática, Univ. Trás-os-Montes e Alto Douro, Vila Real, Portugal*
E. Weinmüller, *Institute for Analysis and Scientific Computing, Vienna University of Technology, Austria*

2000 MSC: 65L05, 34B16

Abstract

Analytical properties of a nonlinear singular second order boundary value problem in ordinary differential equations posed on an unbounded domain for the density profile of the formation of microscopic bubbles in a nonhomogeneous fluid are discussed. Especially, sufficient conditions for the existence of solutions are derived. Two computational methods, a shooting and a collocation method, are proposed for the numerical treatment of the analytical problem. The results of numerical simulations are presented and discussed.

Keywords: Singular boundary value problems, nonlinear ordinary differential equations, degenerate Laplacian, collocation methods, shooting methods.

1 Introduction

In this paper, we consider a second order nonlinear ordinary differential equation arising in the modeling of non-homogeneous fluids. In the Cahn-Hilliard theory for mixtures of fluids (see, for example, [5]) an additional term involving the gradient of the density ($\mathrm{grad}(\rho)$) is added to the classical expression $E_0(\rho)$ for the volume free energy, depending on the density ρ of the medium. Hence the total volume free energy of a non-homogeneous fluid ca be written as

$$E(\rho, \mathrm{grad}(\rho)) = E_0(\rho) + \frac{\sigma}{2}(\mathrm{grad}(\rho))^2, \tag{1.1}$$

where $E_0(\rho)$ is a double-well potential, whose wells define the phases. The potential $E_0(\rho)$ causes an interfacial layer within which the density ρ suffers large variations [9].

Because of the shape of E_0, the fluid tends to divide into two phases with densities $\rho = \rho_l$ (liquid) and $\rho = \rho_v$ (vapour) and the term $\frac{\sigma}{2}(\mathrm{grad}(\rho))^2$ tends to turn the interface between them into a thin layer, endowing it with energy (the surface tension) [26].

When the free energy is given by (1.1) the density profile $\rho(r)$ can be obtained in the stationary case by means of minimization of the functional,

$$J(\rho) = \int_\Omega \left(E_0(\rho) + \frac{\sigma}{2}(\mathrm{grad}(\rho))^2 \right) dr, \tag{1.2}$$

where $\Omega \subset R^N$. This minimization problem leads to a partial differential equation of the form

$$\gamma \nabla \rho = \mu(\rho) - \mu_0, \tag{1.3}$$

where $\mu(\rho) = \frac{dE_0}{d\rho}$ is the chemical potential of the considered mixture of fluids. In the case of spherical symmetry, which is the most common in applications, equation (1.3) can be reduced to a second order ordinary differential equation of the form

$$r^{1-N}(r^{N-1}\rho'(r))' = f(\rho(r)), \quad r > 0, \tag{1.4}$$

where N is the space dimension and f represents the right-hand side of (1.4). This function is usually known and depends on the properties of the considered mixture of fluids. Typically it is a cubic polynomial of ρ, with three real roots. Choosing an adequate system of units, we may write f as

$$f(\rho) := 4\lambda^2 \rho(\rho + 1)(\rho - \xi), \tag{1.5}$$

where $\xi = \rho_v$ (vapour density) and λ is a real parameter. Equation (1.4) is called the density profile equation and was studied, for example in [6, 7]. The authors of those articles show that the density profile (ρ) is a monotone solution of equation (1.4) satisfying the boundary conditions,

$$\rho'(0) = 0, \quad \lim_{r\to\infty} \rho(r) = \xi. \tag{1.6}$$

In [14, 20, 21], a detailed study of the boundary value problem (1.4), (1.6) has been provided. In [20, 21] the asymptotic properties of the solutions near the singular points, 0 and ∞, have been studied. This enables approximate representations of the solution for $r \to 0$ and $r \to \infty$. Based on these representations, stable shooting methods were implemented for the numerical solution of the problem. Moreover, in [14], accurate numerical results were obtained for this problem, using the BVPSUITE code based on polynomial collocation.

In the present paper, we study a generalization of the problem (1.4), (1.6). From the physical point of view, this more general situation arises when the coefficient σ in (1.1) is not constant. Such a model was considered, for example, in [13]. Here, the free volume energy takes a different form,

$$E(\rho, \mathrm{grad}(\rho)) = E_0(\rho) + \frac{c}{p}|\mathrm{grad}(\rho)|^p, \tag{1.7}$$

where c and $p > 1$ are given constants[1]. Such models were analyzed in [24, 25]. Replacing expression (1.1) by (1.7) results in

$$c\,\mathrm{div}(|\mathrm{grad}(\rho)|^{p-2}\mathrm{grad}(\rho)) = \frac{dE_0}{d\rho} - d, \quad d > 0. \tag{1.8}$$

[1]For $p = 2$ and $c = \sigma$ we obtain the simple case (1.1)

The operator on the left-hand side of (1.8) is the so called p-Laplacian. Moreover, since here the case of spherical bubbles is considered, spherical symmetry is used to reduce the dimension of the problem. The study of the more general case, without spherical symmetry, can be found in [24, 25]. Using spherical coordinates the following ordinary differential equation can be derived:

$$cr^{1-N}(|\rho'(r)|^{p-2}\rho'(r))' = \mu(\rho) - \mu(\rho_l). \tag{1.9}$$

Introducing the adequate system of units in (1.9) and denoting $f_p(\rho) := \mu(\rho) - \mu(\rho_l)$ yields then the following equation:

$$r^{1-N}(r^{N-1}|\rho'(r)|^{p-2}\rho'(r))' = f_p(\rho), \quad r > 0. \tag{1.10}$$

The aim of this paper is to analyze the boundary value problem (1.10), (1.6) and propose efficient numerical methods for its approximate solution.

The paper is organized as follows. In Section 2, we discuss existence of solutions to the boundary value problem (1.10). In Section 3, a shooting method based on the asymptotic properties of solutions and the numerical approximation obtained by a collocation method are introduced. Numerical results are presented in Section 4 and conclusions can be found in Section 5.

2 Existence of Solutions to the Analytical Problem

As in the case $p = 2$, the existence analysis is based on the results provided in [8]. In order to apply these results, we have to carry out in (1.10) the following variable substitution:

$$\rho = \xi - u.$$

In the new variable equation (1.10) takes the form

$$r^{1-N}(r^{N-1}|u'(r)|^{p-2}u'(r))' + g_p(u) = 0, \quad r > 0, \tag{2.1}$$

where

$$g_p(u) = f_p(\xi - u).$$

In this case boundary conditions (1.6) are replaced by

$$u'(0) = 0, \quad \lim_{r\to\infty} u(r) = 0. \tag{2.2}$$

Let us also introduce the notation

$$F_p(s) = \int_0^s g_p(u)du. \tag{2.3}$$

According to Theorem 1 in [8] the sufficient conditions for the existence of a solution to the problem

$$r^{1-N}(r^{N-1}|u'(r)|^{p-2}u'(r))' + g_p(u) = 0, \quad r > 0, \quad u'(0) = 0, \quad \lim_{r\to\infty} u(r) = 0 \tag{2.4}$$

are

C1 g_p is locally Lipschitz continuous on $(0, \infty)$ and absolutely integrable on any interval $[0, s]$. Therefore $F_p(s)$ exists for any $s > 0$ and $F_p(0) = 0$.

C2 There exists some $\beta > 0$ such that $F_p(s) < 0$ for $0 < s < \beta$, $F_p(\beta) = 0$, and $g_p(\beta) > 0$.

C3 There exists some $\eta > 0$ such that $g_p(s) \leq 0$ for $0 < s \leq \eta$ and

$$\int_0^a |F_p(s)|^{-1/p} ds = \infty, \tag{2.5}$$

for any positive a and $p > 1$.

C4 There exists some positive k_0 such that

$$\lim_{s \to \gamma-} sup \frac{g_p(s)}{(\gamma - s)^{p-1}} < k_0, \tag{2.6}$$

where γ is the least zero of g_p greater than β.

We now define g_p in such a way that these conditions are satisfied. First of all, since g_p must have the same roots and the same sign as $g_2(\rho)$ for any value of p, we look for g_p in the form

$$g_p(u) = 2p\lambda^2(u - \xi - 1)(\xi - u)u|u - \xi - 1|^\alpha |u|^\alpha, \tag{2.7}$$

where $p > 1$ and α is an adjustable positive parameter[2].

Since $g_p(0) = 0$ and g_p is differentiable at the origin, it follows from (2.3) and (2.7) that F_p has a root of multiplicity larger or equal to two at the origin. Moreover, this function has two real roots s_1, s_2, such that $\xi < s_1 < \xi + 1 < s_2$, see Figure 1 and Figure 2. Hence, F_p can be written in the form

$$F_p(s) = -\lambda^2 s^2 |s|^\alpha P(s), \tag{2.8}$$

where λ is a constant and P is a smooth function such that $P(0) \neq 0$. This means that P has roots only at s_1 and s_2. First of all, we note that due to (2.7), g_p satisfies the condition C1, for any nonnegative value of α. Moreover, as it was pointed out in [21], for $p = 2$ and $0 < \xi < 1$, condition C2 is satisfied with $\beta = s_1$, where s_1 is the least positive root of F. The same holds for any $p > 1$ if $\alpha \geq 0$.

Concerning conditions C3 and C4, we have to choose proper values for α in (2.7) and (2.8) in order to assure that also these conditions are satisfied. Let us begin with condition C3. Due to the form of F_p given by (2.8), we conclude that

$$|F(s)|^{-1/p} = |\lambda|^{-2/p} |s|^{-\frac{\alpha+2}{p}} |P(s)|^{-\frac{1}{p}}.$$

Therefore, the integral (2.5) diverges if $\alpha \geq p - 2$ and converges if $\alpha < p - 2$. Hence, we choose $\alpha \geq p - 2$ for the condition C3 to hold. Moreover, note that in

[2] In case $p = 2$, we set $\alpha = 0$ and have $f_2 = f$.

Figure 1: Graphs of g and F for the case $p = 2$

our problem g_p has a unique root greater than β, which is $\xi + 1$. Hence, we have $\gamma = \xi + 1$. Therefore, we see that the limit (2.6) is zero for $\alpha > p - 2$, positive for $\alpha = p - 2$, and infinite for $\alpha < p - 2$. We conclude that the condition C4 is also satisfied for $\alpha \geq p - 2$.

Using results given in [8], we can guarantee that for g_p of the form (2.7), with $\alpha \geq p - 2$ and $0 < \xi < 1$, there exists at least one monotone nontrivial solution of the boundary value problem (2.4). We now consider the variable $\rho = \xi - u$. In order to assure that the problem (1.10), (1.6) has at least one solution, the function f_p has to be of the form,

$$f_p(\rho) = 2p\lambda^2(\rho - \xi)(\rho + 1)\rho|\rho - \xi|^\alpha|\rho + 1|^\alpha, \tag{2.9}$$

where ξ and α satisfy the above conditions. In the sequel, we choose $\alpha = p - 2$ for $p \geq 2$. The case $p < 2$ will not be considered in the present work. Therefore, taking into account that $-1 < \rho < \xi$ for $p > 2$, we can consider f_p in the simplified form

$$f_p(\rho) = -2p\lambda^2|\rho - \xi|^{p-1}\rho(\rho + 1)^{p-1}. \tag{2.10}$$

In Figure 1 the graphs of g and F are shown for $p = 2$. The graphs of g_p and F_p for $p = 3$, $\alpha = 1$, are given in Figure 2.

3 Numerical Approximation

3.1 Shooting Method

In order to design a suitable shooting method it was necessary to accordingly extend the algorithm described in [21]. The main idea is to replace the con-

Figure 2: Graphs of g_p and F_p for the case $p = 3$

sidered boundary value problem with two singularities, at the origin and at infinity, by two auxiliary boundary value problems, each of them having only one singularity. We now describe the method in more detail.

Step 1. We start by fixing the values of r_0, δ, and r_∞, in such a way that $r_\infty > r_0 > \delta > 0$. Here, r_0 is an initial approximation to the root of the solution, the so-called bubble radius, δ is close to zero and r_∞ is large enough for the asymptotical approximation of the solution at infinity, obtained in [**?**], to be related to this point. Next, we subdivide the region $[\delta, r_\infty]$ in which the approximate solution will be provided into two subintervals, $[\delta, r_0]$ and $[r_0, r_\infty]$. Let $\rho_-(r)$ be a monotone solution of (1.10) on $[\delta, r_0]$ which satisfies the boundary conditions

$$\rho_-(\delta) = \rho_0 + \frac{p-1}{p}\left(\frac{f_p(\rho_0)}{N}\right)^{\frac{1}{p-1}} \delta^{\frac{p}{p-1}} \left(1 + y_1 \delta^{\frac{p}{p-1}}\right), \quad \rho_-(r_0) = 0. \tag{3.1}$$

Analogously, let $\rho_+(r)$ be a monotone solution of (1.10) on $[r_0, \infty)$ satisfying the boundary conditions

$$\rho_+(r_0) = 0, \quad \rho_+(r_\infty) = \xi - b r_\infty^a C_1(r_\infty) e^{-\tau r_\infty}. \tag{3.2}$$

For more details on (3.1) and (3.2), especially on function C_1 and constants y_1, a, b, and τ, which follow from asymptotic considerations, we refer the reader to [22].
Finally, let us define

$$\rho(r) = \begin{cases} \rho_-(r), \text{ if } \delta \le r \le r_0, \\ \rho_+(r), \text{ if } r_0 \le r \le r_\infty. \end{cases} \tag{3.3}$$

Note that the functions ρ_- and ρ_+ are obtained by the standard shooting method. This means that we first compute ρ_0 such that ρ_- is a solution of the first, and b such that ρ_+ is a solution of the second auxiliary problem. When applying the shooting method we have to take into account that ρ_0 in (3.1) has to satisfy $-1 < \rho_0 < 0$ and b in (3.2) has to be positive.

Step 2. In general, the function $\rho(r)$ from (3.3) is not a solution of (1.10), (1.6) on $[0, \infty[$ because the condition

$$\lim_{r \to r_0-} \rho'(r) = \lim_{r \to r_0+} \rho'(r) \tag{3.4}$$

is not satisfied for the given r_0. Let us compute the difference

$$\Delta(r_0) = \lim_{r \to r_0-} \rho'(r) - \lim_{r \to r_0+} \rho'(r). \tag{3.5}$$

Our goal is now to find such a value $r_0 \in \mathbb{R}_+$ that $\Delta(r_0) = 0$. In order to find the required value of r_0 we use the secant method, starting from two values r_a and r_b, such that $\Delta(r_a) < 0$ and $\Delta(r_b) > 0$ holds, respectively.

Step 3. In order to extend the approximate solution to the intervals $[0, \delta]$ and $[r_\infty, \infty)$, we use the asymptotic expansions derived in [22]. In Section 4, we shall present numerical results obtained from the algorithm described above.

The method has been implemented using Mathematica. This enables to compute the approximations accurately within a reasonable computational time for a large range of values of p and ξ. The advantage of the proposed approach is that it makes use of the asymptotic solution properties near the singular points. In the present implementation, the auxiliary problems for ρ_- and ρ_+ with boundary conditions (3.1) and (3.2), respectively, are solved using the package NDSOLVE in Mathematica [27]. The drawback of this method is that one can neither control the global error of the approximation nor the required total computational effort. In Section 3.2, we describe a different approach to the numerical solution of the problem (1.10), (1.6). This approach is based on polynomial collocation and makes use of the recently developed Matlab code `bvpsuite`, cf. [15], designed for the numerical solution of boundary value problems in ordinary differential equations, posed on unbounded domains.

3.2 Collocation Method

The motivation for the development of the Matlab code `bvpsuite` [15] were singular boundary value problems of the form

$$z'(t) = \frac{M(t)}{t^\alpha} z(t) + f(t, z(t)), \quad t \in (0, 1], \tag{3.6}$$

$$B_0 z(0) + B_1 z(1) = \beta, \tag{3.7}$$

where $\alpha \geq 1$, z is an n-dimensional real function, M is a smooth $n \times n$ matrix and f is an n-dimensional smooth function on a suitable domain. B_0 and B_1 are constant matrices which are subject to certain restrictions for a well-posed

problem. (3.6) is said to feature a *singularity of the first kind* for $\alpha = 1$, while for $\alpha > 1$ the problem has a *singularity of the second kind*, also commonly referred to as *essential singularity*. The analytical properties of the problem (3.6) have been discussed in [10, 12], with a special focus on the most general boundary conditions which guarantee well-posedness of the problem. To compute the numerical solution of (3.6) we use polynomial collocation. Our decision to use polynomial collocation was motivated by its advantageous convergence properties for (3.6), while in the presence of a singularity other high order methods show order reductions and become inefficient. In [3, 11, 18] convergence results for collocation applied to problems with a singularity of the first kind, $\alpha = 1$, were shown. The usual high-order superconvergence at the mesh points does not hold in general for singular problems, however, the uniform superconvergence is preserved (up to logarithmic factors), see [18] for details.

Motivated by these observations, we have implemented two MATLAB codes for singular boundary value problems. For higher efficiency, we provide an estimate of the global error and adaptive mesh selection. Transformation of problems posed on semi-infinite intervals to $[0, 1]$ makes the solution of such problems also accessible by our methods. All these algorithmic components have been integrated into two MATLAB codes. `sbvp` solves explicit first order ODEs [1], while `bvpsuite` can be applied to arbitrary order problems also in implicit formulation and differential algebraic equations [19]. Moreover, a pathfollowing strategy extends the scope of the latter code [17].

3.2.1 Basic Solver in the MATLAB Code `bvpsuite`

The code is designed to solve systems of differential equations of arbitrary mixed order including zero[3], subject to initial or boundary conditions,

$$\begin{aligned} &F\big(t, p_1, \dots, p_s, z_1(t), z_1'(t), \dots, z_1^{(l_1)}(t), \dots, z_n(t), z_n'(t), \dots, z_n^{(l_n)}(t)\big) = 0, \\ &B\big(p_1, \dots, p_s, z_1(c_1), \dots, z_1^{(l_1-1)}(c_1), \dots, z_n(c_1), \dots, z_n^{(l_n-1)}(c_1), \dots, \\ &\qquad\qquad z_1(c_q), \dots, z_1^{(l_1-1)}(c_q), \dots, z_n(c_q), \dots, z_n^{(l_n-1)}(c_q)\big) = 0, \end{aligned} \tag{3.8}$$

where the solution $z(t) = (z_1(t), z_2(t), \dots, z_n(t))^T$, and the parameters p_i, $i = 1, \dots, s$, are unknown. In general, $t \in [a, b], -\infty < a, b < \infty$[4]. Moreover, $F : [a, b] \times \mathbb{R}^s \times \mathbb{R}^{l_1} \times \cdots \times \mathbb{R}^{l_n} \to \mathbb{R}^n$ and $B : \mathbf{R}^s \times \mathbf{R}^{ql_1} \times \cdots \times \mathbf{R}^{ql_n} \to \mathbf{R}^{l+s}$, where $l := \sum_{i=1}^n l_i$. Note that boundary conditions can be posed on any subset of distinct points $c_i \in [a, b]$, with $a \le c_1 < c_2 < \cdots < c_q \le b$. For the numerical treatment, we assume that the boundary value problem (3.8) is well-posed and has a locally unique solution z.

In order to find a numerical solution of (3.8) we consider a mesh $[\tau_i, \tau_{i+1}]$, $i = 0, \dots, N-1$, partitioning the interval $[a, b]$. Every subinterval $[\tau_i, \tau_{i+1}]$ contains m collocation points $t_{i,j}$, $j = 1, \dots, m$. Let $\mathbf{P}_m$ be the space of piecewise

[3] This means that differential-algebraic equations are also in the scope of the code.
[4] For the extension to unbounded domains, see Section 3.2.4.

polynomial functions of degree $\leq m$, which are globally continuous in $[a, b]$. In every subinterval J_i we make an ansatz $P_{i,k} \in \mathbf{P}_{m+l_k-1}$ for the k-th solution component z_k, $k = 1, \ldots, n$, of the problem (3.8). In order to compute the coefficients in the ansatz functions we require that (3.8) is satisfied exactly at the collocation points. Moreover we require that the collocation polynomial $p(t) := P_i(t)$, $t \in J_i$, is a globally continuous function on $[a, b]$ with components in $C^{l_i-1}[a, b]$, $i = 1, \ldots, n$, and that the boundary conditions hold. All these conditions imply a nonlinear system of equations for the unknown coefficients in the ansatz function. For more details see [16].

3.2.2 Error Estimates for the Global Error of the Collocation

To provide an asymptotically correct estimate for the global error of the collocation solution we propose to use the classical error estimate based on mesh halving. In this approach, we compute the collocation solution on a mesh $\Delta := \{\tau_i, i = 0, \ldots, N\}$ with step sizes h_i and denote this approximation by $p_\Delta(t)$. Subsequently, we choose a second mesh Δ_2 where in every interval of Δ we insert two subintervals of length $h_i/2$. On this mesh, we compute the numerical solution based on the same collocation scheme to obtain the collocating function $p_{\Delta_2}(t)$. Using these two quantities, we define $\mathcal{E}(t) := 2^m(p_{\Delta_2}(t) - p_\Delta(t))/(1-2^m)$ as an error estimate for the approximation $p_\Delta(t)$. Generally, estimates of the global error based on mesh halving work well for both problems with a singularity of the first kind and for essentially singular problems [2]. Since they are also applicable to higher-order problems and problems in implicit form (as for example DAEs) without the need for modifications, we have implemented this strategy in our code `bvpsuite`.

3.2.3 Adaptive Mesh Selection

The mesh selection strategy implemented in `bvpsuite` was proposed and investigated in [23]. Most modern mesh generation techniques in two-point boundary value problems construct a smooth function mapping a uniform auxiliary grid to the desired nonuniform grid. In [23] a new system of control algorithms for constructing a grid density function $\phi(t)$ is described. The local mesh width $h_i = \tau_{i+1} - \tau_i$ is computed as $h_i = \epsilon_N/\varphi_{i+1/2}$, where $\epsilon_N = 1/N$ is the accuracy control parameter corresponding to $N - 1$ interior points, and the positive sequence $\Phi = \{\varphi_{i+1/2}\}_{i=0}^{N-1}$ is a discrete approximation to the continuous density function $\phi(t)$, representing the mesh width variation. Using an error estimate, a feedback control law generates a new density from the previous one. Digital filters may be employed to process the error estimate as well as the density.

For boundary value problems, an adaptive algorithm must determine the sequence $\Phi^{[\nu]}$ in terms of problem or solution properties. True adaptive approaches equidistribute some *monitor function*, a measure of the residual or error estimate, over the interval. As $\Phi^{[\nu]}$ will depend on the error estimates, which in turn depend on the distribution of the grid points, the process of finding the density becomes *iterative*. For some error control criteria a local grid

change typically has global effects. The techniques developed here avoid this difficulty by restricting the error estimators to those having the property that the estimated error on the interval $[\tau_i, \tau_{i+1}]$ only depends on the local mesh width, $h_i = \epsilon_N/\varphi_{i+1/2}$.

In order to be able to generate the mesh density function, we decided to use the residual $r(t)$ to define the monitor function. The values of $r(t)$ are available from the substitution of the collocation solution $p(t)$ into the analytical problem (3.8). In the grid adaptation routine implemented in the code mesh generation, finding the optimal density function, is separated from mesh refinement, finding the proper number of mesh points. We first try to provide a good density function Φ on a rather coarse mesh with a fixed number of points $M = 50$. The mesh density function is chosen to equidistribute the monitor function. For each density profile in the above iteration, we then estimate the number of mesh points necessary to reach the tolerance.

3.2.4 Semi-Infinite Intervals

Our code can also treat problems posed on semi-infinite intervals $t \in (a, \infty),\ a > 0$ (and by a splitting of the interval, also $a = 0$). In order to exploit our efficient and robust mesh selection strategy also in this case, we use the transformation $t = \frac{1}{\tau},\ z(t) = x\left(\frac{1}{\tau}\right)$ to restate $x'(\tau) = \tau^\beta f(\tau, x(\tau))$, $\tau \in [a, \infty)$, $\beta > -1$ as the problem $z'(t) = -f(1/t, z(t))/t^{\beta+2}$, $t \in (0, 1/a]$. This is in general a problem with an essential singularity, which however is in the scope for our collocation methods, error estimation procedure and adaptive mesh refinement. In this approach, the mesh is adapted only according to the unsmoothness of the solution without the need for mesh grading on long intervals, and moreover no truncation of the unbounded interval is necessary. This strategy was employed successfully for example in [4, 14].

3.2.5 Reformulation of the Original Problem for the Collocation Method

We now solve the problem (1.10), (1.6) using the collocation method implemented in `bvpsuite`. Here, the original differential equation is transformed to a system of four implicit first order differential equations. The main idea is to split the unbounded domain $(0, \infty)$ the original differential equation is posed on, into $(0, 1)$ and $[1, \infty)$.

For $r \in (0, 1)$, we define the independent and dependent variables as $t := r$ and $z_1(t) := \rho(r)$, respectively. Let us introduce

$$z_3(t) := |z_1'(t)|^{p-2} z_1'(t). \tag{3.9}$$

Then, for $z_3 \in C^1(0, 1)$, it follows immediately from (1.10),

$$z_3'(t) + \frac{N-1}{t} z_3(t) = f_p(z_1(t)). \tag{3.10}$$

For $r \in [1,\infty)$ the new variables are $t := \frac{1}{r}$ and $z_2(t) := \rho\left(\frac{1}{r}\right)$. Note that $\frac{d}{dr} = \frac{d}{dt}(-t^2)$ and $\frac{d}{dt} = \frac{d}{dr}\left(-\frac{1}{t^2}\right)$.
Moreover,

$$\frac{dz_2}{dt}(t) = \frac{d\rho}{dr}(r)\left(-\frac{1}{t^2}\right). \tag{3.11}$$

Using (3.11), we rewrite (1.10) and obtain

$$t^{N-1}\frac{d}{dr}\left(t^{1-N}\left|\frac{dz_2}{dt}(t)(-t^2)\right|^{p-2}\frac{dz_2}{dt}(t)(-t^2)\right) = f_p(z_2(t)), \quad t \in (0,1). \tag{3.12}$$

Introducing $z_4(t) := \left|\frac{dz_2}{dt}(t)\right|^{p-2}\frac{dz_2}{dt}(t)$ and assuming $z_4 \in C^1(0,1]$, equation (3.12) can be written in the following form:

$$t^{N-1}\frac{d}{dr}\left(-t^{2p-N-1}z_4(t)\right) = t^{N-1}\frac{d}{dt}\left(-t^{2p-N-1}z_4(t)\right)(-t^2) = f_p(z_2(t)). \tag{3.13}$$

This yields the following system of four implicit differential equations posed on $(0,1]$,

$$(2p-N-1)t^{2p-1}z_4(t) + t^{2p}z_4'(t) = f_p(z_2(t)), \tag{3.14}$$

$$z_4(t) = |z_2'(t)|^{p-2}\,z_2'(t), \tag{3.15}$$

$$z_3'(t) + \frac{N-1}{t}z_3(t) = f_p(z_1(t)), \tag{3.16}$$

$$z_3(t) = |z_1'(t)|^{p-2}z_1'(t). \tag{3.17}$$

For the above system four boundary conditions are required. From (1.6) we have

$$z_3(0) = 0, \quad z_2(0) = \xi.$$

Furthermore, both solution branches z_1 and z_2 shall match in $t = 1$ in such a way that $\rho \in C^1(0,\infty)$ holds. Consequently, again using (3.11), we require,

$$z_1(1) = z_2(1), \quad z_3(1) = -z_4(1).$$

4 Numerical Results

In this section, we present numerical results obtained by the shooting and collocation methods described above. The main purpose of the numerical simulation is to test and compare the efficiency and robustness of the proposed algorithms.

We begin by presenting numerical results which show how the physical properties of the bubbles depend on p and ξ. Besides the bubble radius R and the gas density at the center of the bubble, we will also calculate the numerical values of the energy integral J defined by

$$J := J(\rho) := \int_0^\infty \left(\frac{\rho'(r)^p}{p} + W_p(\rho(r))\right) r^{N-1}dr, \tag{4.1}$$

where

$$W_p(\rho) = \int_0^\rho g_p(u)du. \tag{4.2}$$

The solution of the boundary value problem can be viewed as the function ρ for which the integral J is minimized.

In order to evaluate J, we use a numerical quadrature method. Keeping in mind that we use a collocation method to solve (3.14)–(3.17), collocation points could be used as mesh points in the involved quadrature rule. We now introduce the partition $\Delta_\mathcal{T}$ of the interval of integration $[0,1]$,

$$\Delta_\mathcal{T} = \{0 = \tau_0 < \tau_1, \ldots, \tau_j < \tau_{j+1}, \ldots, \tau_N = 1, \;\; h_j := \tau_{j+1} - \tau_j\}.$$

In each subinterval $[\tau_j, \tau_{j+1}]$, we specify m collocation points $t_{j,l}$, $l = 1, 2, \ldots, m$, such that

$$\tau_j < t_{j,1} < t_{j,2} < \ldots < t_{j,m} < \tau_{j+1},$$

cf. Section 3.2.1.

Let us denote by $P_{j,k}(t)$ the collocation polynomial approximating $z_k(t)$ for $t \in [\tau_j, \tau_{j+1}]$, $k = 1, \ldots, 4$, $j = 0, 1, \ldots, N-1$. Then, we can approximate the values of the first derivative $z_k'(t)$ at the collocation points $t_{j,l}$, $j = 0, 1, \ldots, N-1$, $l = 1, 2, \ldots, m$, directly from the definition of the collocation method. Since z_1 is monotonically increasing, we conclude from (3.17),

$$z_1'(t) = z_3(t)^{\frac{1}{p-1}},$$

and consequently,

$$P_{j,1}'(t_{j,l}) = P_{j,3}(t_{j,l})^{\frac{1}{p-1}}, \quad j = 0, 1, \ldots, N-1, \;\; l = 1, 2, \ldots, m. \tag{4.3}$$

For the second component we obtain a very similar relation. Due to (3.11), z_2 is monotonically decreasing, and thus from (3.15) it follows $|z_2'(t)| = z_4(t)\,|z_4(t)|^{\frac{1}{p-1}}$ which together with $|z_2'| = -z_2'$ yields

$$(-P_{j,2}'(t_{j,l}))^p = |P_{j,4}(t_{j,l})|^{\frac{p}{p-1}}, \quad j = 0, 1, \ldots, N-1, \;\; l = 1, 2, \ldots, m. \tag{4.4}$$

Let us assume that a numerical quadrature rule is defined on the interval $[0,1]$ via the weights $\{w_l\}_{l\in\{1,\ldots,n\}}$ and the evaluation points $\xi_l \in [0,1]$, $l = 1, \ldots, n$,

$$\int_0^1 f(t)dt \approx Q_{[0,1]}(f) = \sum_{l=1}^n w_l f(\xi_l).$$

In order to integrate over the interval $[\tau_j, \tau_{j+1}]$ we have to recalculate the weights and the evaluation points accordingly,

$$\int_{\tau_j}^{\tau_{j+1}} f(t)dt \approx Q_{[\tau_j,\tau_{j+1}]}(f) = h_j \sum_{l=1}^n w_l f((\tau_{j+1} - \tau_j)\xi_l + \tau_j).$$

We first rewrite J,

$$
\begin{aligned}
J(\rho) &= \int_0^1 \left(\frac{1}{p}\rho'(r)^p + W_p(\rho(r))\right) r^{N-1}\,dr \\
&\qquad\qquad + \int_1^\infty \left(\frac{1}{p}\rho'(r)^p + W_p(\rho(r))\right) r^{N-1}\,dr \\
&= \int_0^1 \left(\frac{1}{p}z_1'(t)^p + W_p(z_1(t))\right) t^{N-1}\,dt \\
&\qquad\qquad + \int_0^1 \left(\frac{1}{p}\frac{(-z_2'(t))^p}{t^{N+1+2p}} + \frac{W_p(z_2(t))}{t^{N+1}}\right) dt.
\end{aligned}
$$

Choosing a Gaussian quadrature rule and the collocation scheme with $m = n$ and $(\tau_{j+1} - \tau_j)\xi_l + \tau_j = t_{j,l}$ enables to use (4.3) and (4.4). Thus, we obtain

$$
\begin{aligned}
J(\rho) &\approx Q_{[0,1]}\left(\left(\frac{1}{p}z_1'(t)^p + W_p(z_1(t))\right) t^{N-1} + \frac{1}{p}\frac{(-z_2'(t))^p}{t^{N+1+2p}} + \frac{W_p(z_2(t))}{t^{N+1}}\right) \\
&= \sum_{\substack{j\in\{0,1,\dots,N-1\}\\ l\in\{1\dots m\}}} w_l \left(\left(\frac{1}{p}z_1'(t_{j,l})^p + W_p(z_1(t_{j,l}))\right) t_{j,l}^{N-1}\right. \\
&\qquad\qquad \left. + \frac{1}{p}\frac{(-z_2'(t_{j,l}))^p}{t_{j,l}^{N+1+2p}} + \frac{W_p(z_2(t_{j,l}))}{t_{j,l}^{N+1}}\right) \\
&\approx \sum_{\substack{j\in\{0,1,\dots,N-1\}\\ l\in\{1\dots m\}}} w_l \left(\left(\frac{\left(P_{j,1}'(t_{j,l})\right)^p}{p} + W_p(P_{j,1}(t_{j,l}))\right) t_{j,l}^{N-1}\right. \\
&\qquad\qquad \left. + \frac{\left(-P_{j,2}'(t_{j,l})\right)^p}{p\,t_{j,l}^{N+1+2p}} + \frac{W_p(P_{j,2}(t_{j,l}))}{t_{j,i}^{N+1}}\right) \\
&= \sum_{\substack{j\in\{0,1,\dots,N-1\}\\ l\in\{1\dots m\}}} h_j w_l \left(\left(\frac{P_{j,3}(t_{j,l})^{\frac{p}{p-1}}}{p} + W_p(P_{j,1}(t_{j,l}))\right) t_{j,l}^{N-1}\right. \\
&\qquad\qquad \left. + \frac{|P_{j,4}(t_{j,l})|^{\frac{p}{p-1}}}{p\,t_{j,l}^{N+1+2p}} + \frac{W_p(P_{j,2}(t_{j,l}))}{t_{j,l}^{N+1}}\right)
\end{aligned}
$$

Note that in the last sum only values of the solution at the collocation points provided by `bvpsuite` occur and therefore the evaluation of J is straightforward.

In Tables 1, 2, and 3, we present the numerical values of J, R, and ρ_0 for $p = 3$, $p = 3.5$, and $p = 4$, respectively. We also give an error estimate for the approximation of J. The results reported in the tables are provided for the case $N = 3$ and $\xi = 0.1, 0.2, \dots, 0.8$. They were obtained using `bvpsuite` based on

polynomial collocation. The application of the code is in no way straightforward and it takes quite an effort to calculate the solution with reasonable accuracy. The reason for the difficulties can be explained by the fact that the analytical solution is unsmooth and that condition numbers of the involved linear systems of equations arising during the Newton iteration become significant for $h \to 0$. This means that the starting values for the Newton procedure have to be very accurate. Clearly, we can use the solution profiles obtained for a certain value of ξ as a starting guess for the next value $\xi + \Delta_\xi$, but for the Newton method to work Δ_ξ has to be small. Moreover, to recover correctly the solution behavior in the region where its profile is very steep, very fine grids are necessary. This results in high demand for computational recourses (storage and time).

ξ	J	err_J	ρ_0	R	err_{sol}
0.1	0.0008400	5.25e-08	-0.1553002	2.1128538	1.92e-05
0.2	0.0079429	4.69e-07	-0.3196202	1.8217471	3.45e-05
0.3	0.0324557	2.15e-06	-0.4899207	1.7632653	5.58e-05
0.4	0.0965061	8.96e-06	-0.6588715	1.8248666	8.77e-05
0.5	0.2497196	3.75e-05	-0.7650025	2.0303955	2.20e-04
0.6	0.6205862	2.17e-04	-0.9259004	2.3608133	7.87e-04
0.7	1.6241505	1.72e-03	-0.9850873	3.0318875	4.22e-03
0.8	5.1044359	2.75e-02	-0.9994747	4.4949907	4.29e-02

Table 1: Numerical results for $p = 3$.

ξ	J	err_J	ρ_0	R	err_{sol}
0.1	0.0001225	2.73e-09	-0.1253690	1.731092	1.04e-05
0.2	0.0018308	3.56e-08	-0.2622161	1.534213	1.58e-05
0.3	0.0097973	2.43e-07	-0.4104270	1.501617	2.22e-05
0.4	0.0352165	1.38e-06	-0.5677791	1.557476	3.77e-05
0.5	0.1050056	7.75e-06	-0.7265417	1.705349	7.27e-05
0.6	0.2916065	4.92e-05	-0.8679810	1.991735	2.93e-04
0.7	0.8328526	4.28e-04	-0.9630508	2.538646	1.74e-03
0.8	2.8107513	7.30e-03	-0.9973830	3.727937	1.90e-02

Table 2: Numerical results for $p = 3.5$.

In Table 4, we illustrate the convergence order of the collocation method and the condition numbers of the matrices arising during the Newton iteration. These results show clearly that the problem is ill-conditioned and a drop of the convergence order can be observed, due to the unsmoothness of the solution at the origin. In case of an appropriately smooth solution, the convergence order would be two, but in the present case the second derivative of the solution is unbounded as $r \to 0$ for $p > 0$, cf.[22].

It is interesting to see that for a fixed value of ξ, the condition numbers

ξ	J	err_J	ρ_0	R	err_{sol}
0.1	0.0000200	1.01e-10	-0.105231627	1.4527009	1.23e-05
0.2	0.0004610	1.30e-09	-0.222392833	1.3167929	1.63e-05
0.3	0.0031933	4.20e-09	-0.352719668	1.3023511	2.01e-05
0.4	0.0137922	1.29e-07	-0.496736196	1.3559939	2.84e-05
0.5	0.0473178	1.36e-06	-0.651939112	1.4827001	4.27e-05
0.6	0.1467129	1.18e-05	-0.806737348	1.7228001	7.21e-05
0.7	0.4575437	1.22e-04	-0.931945341	2.1819640	5.69e-04
0.8	1.6592987	2.31e-03	-0.992361016	3.1813922	8.53e-03

Table 3: Numerical results for $p = 4$.

of the matrices grow as p decreases, which can be explained by the fact that for smaller p the interior layer of the solution near the bubble radius becomes steeper. On the other hand, if we fix p, the condition numbers grow as ξ tend to 1. This is well illustrated by the graphs of the solutions in Figure 3. Note the special case $p = 2$ [14]: Here, although the interior layer is steeper when compared with larger values of p, the accuracy of the results is satisfactory since in this case the solution is smooth at the origin, see Figure 3.

h	err_{z1}	ord	err_{z2}	ord	ρ_0	R	cond	ord
2^{-3}	7.57e-02	2.37	8.23e−02	2.45	-0.82852	2.15258	5.42e+06	-1.53
2^{-4}	1.46e-02	2.07	1.51e−02	2.08	-0.77868	2.05249	1.57e+07	-0.03
2^{-5}	3.50e-03	2.02	3.58e−03	2.02	-0.76821	2.03516	1.60e+07	-3.19
2^{-6}	8.64e-04	2.00	8.84e−04	2.00	-0.76565	2.03124	1.46e+08	-2.88
2^{-7}	2.15e-04	2.00	2.20e−04	2.00	-0.76500	2.03040	1.07e+09	-3.53
2^{-8}	5.38e-05	2.00	5.50e−05	2.00	-0.76484	2.03013	1.24e+10	-3.90
2^{-9}	1.35e-05	-	1.37e−05	-	-0.76480	2.03006	1.85e+11	-4.19

Table 4: Numerical results for $(p, \xi) = (3, 0.5)$. Here, err_ρ is an error estimate for the absolute global error of the collocation solution, ord is an estimate of the global convergence order, and cond is the matrix condition number.

Next we present results obtained by the shooting approach. In Table 5, we show the values of the bubble radius R for different values of p and ξ. The corresponding values of ρ_0 can be found in Table 6. The values of the parameter b are given in Table 7. In the computations by the shooting method we used values of δ between $\delta = 0.001$, for the smallest value of p and $\delta = 0.02$, for the largest value of p. The value of r_∞ was set to 6. The values of b strongly depend on the latter parameter. In the tables, for the entries denoted with (*) no reliable results could be provided.

All computations were carried out with the MATLAB precision of 16 digits. However, due to the ill-posedness of the problem, in some cases, especially for ξ close to 1, the results show no more than four correct digits. For the shooting

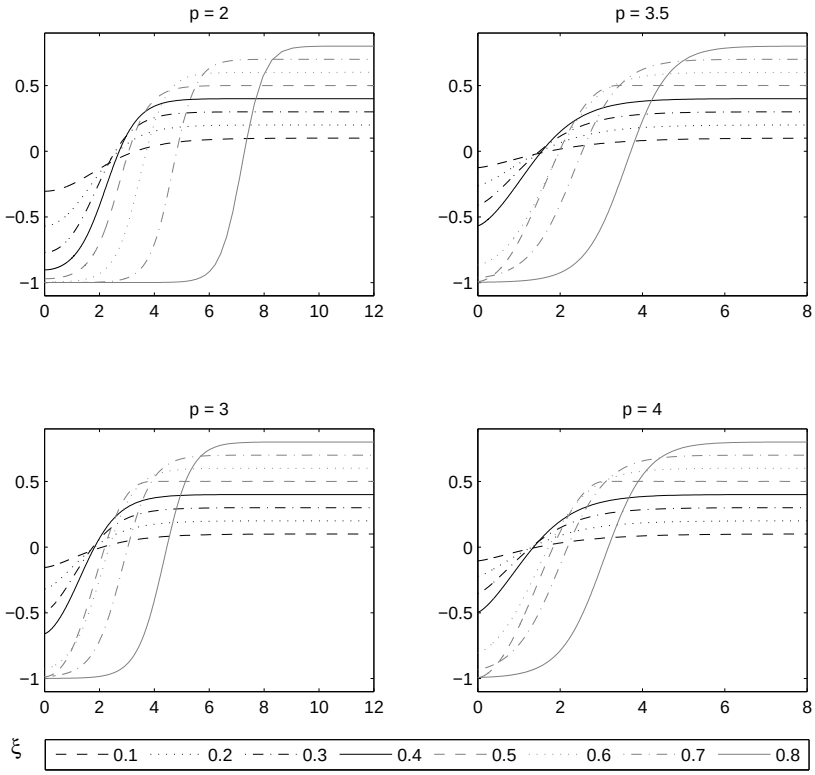

Figure 3: Solution graphs, $\rho(r)$, for different values of p.

method the global error can be only estimated by comparing it to the results obtained by collocation.

p	2.0	2.5	3.0	3.5	4.0
ξ =0.2	2.685	2.20455	1.82180	1.53429	(*)
ξ =0.3	2.582	2.11351	1.76328	1.50167	1.30245
ξ =0.4	2.721	2.19287	1.82478	1.55751	1.35611
ξ =0.5	3.068	2.43240	2.0062	1.70529	1.48283
ξ =0.6	3.696	2.88981	2.35947	1.99133	1.72289
ξ =0.7	4.833	3.73350	3.02514	2.53590	2.18120

Table 5: Approximations of R provided by the shooting method.

p	2.0	2.5	3.0	3.5	4.0
ξ =0.2	-0.5681	-0.40926	-0.31950	-0.26175	(*)
ξ =0.3	-0.7707	-0.60412	-0.489805	-0.409134	-0.350480
ξ =0.4	-0.90313	-0.77370	-0.65871	-0.56697	-0.494054
ξ =0.5	-0.97112	-0.89977	-0.81141	-0.72578	-0.649328
ξ =0.6	-0.99531	-0.97121	-0.92550	-0.86734	-0.804794
ξ =0.7	-0.99979	-0.99646	-0.98392	-0.96266	-0.930961

Table 6: Approximations of ρ_0 provided by the shooting method.

p	2.0	2.5	3.0	3.5	4.0
ξ =0.2	9.857	0.89645	0.238773	0.095496	(*)
ξ =0.3	28.113	1.83466	0.430397	0.167707	0.08531
ξ =0.4	95.98	4.20671	0.838803	0.301533	0.14646
ξ = 0.5	491.92	12.3603	1.9232	0.605841	0.27182
ξ = 0.6	5080.9	57.632	6.14021	1.56118	0.61222
ξ = 0.7	207675	672.673	38.8392	6.88547	2.1355

Table 7: Numerical values of b provided by the shooting method.

We know from the previous results for $p = 2$, see [20, 21], that the bubble radius increases for $\xi \to 0$ and $\xi \to 1$, reaching its minimal value for $\xi \approx 0.28$. According to Figure 4, similar behavior of the bubble radius is observed for larger values of p. On the other hand, for all considered values of p, the density ρ_0 tends to -1 for $\xi \to 1$, cf. Figure 5.

Figure 4: The dependence of the bubble radius R on ξ, for $p = 2$ (bright gray), $p = 3$ (black), and $p = 4$ (dark gray).

Figure 5: The dependence of ρ_0 on ξ, for $p = 2$ (bright gray), $p = 3$ (black), and $p = 4$ (dark gray).

5 Conclusions

A generalization of the density profile equation for nonhomogeneous fluids has been analyzed. Here, the classical Laplacian was replaced by a degenerate one. The right-hand side of the considered differential equation has been defined in such a way that the boundary value problem under consideration has at least one strictly monotone solution. Numerical simulations suggest that some of the main properties of the solution known from the case $p = 2$, are also preserved for other values of p, $2 < p < 4$.

In the present implementation, the shooting method provides good approximations for $p \in [2, 4]$ and $\xi \in [0.2, 0.7]$, except for $p = 4$, $\xi = 0.2$. For values of ξ smaller than 0.2 or greater than 0.7, it is difficult to find the correct values of the shooting parameter ρ_0, because small changes of this parameter often result in a blow-up or a nonmonotone solution.

For the collocation method, the p-Laplacian case is more difficult to handle than the case $p = 2$. This follows from the fact that for $p > 2$ the second and higher derivatives of the solution become unbounded at the origin which makes the adaptive mesh selection inapplicable. Moreover, for values of p in the range $2 < p < 3$ the condition numbers of the matrices involved in the Newton iteration become very large and consequently, numerical approximations become less accurate. On the other hand, the collocation method shows a very good performance for moderate values of ξ.

A different variant of the shooting method has been implemented in [22], where the approximate solutions of the involved initial value problems were obtained from a different solver. In the future, we plan to apply a smoothing variable substitution, resulting in moderate higher solution derivatives near the origin which in turn shall improve the performance of the collocation method.

References

[1] W. Auzinger, G. Kneisl, O. Koch, and E. Weinmüller, *A collocation code for boundary value problems in ordinary differential equations*, *Numer. Algorithms*, 33, 27-39 (2003).

[2] W. Auzinger, O. Koch, D. Praetorius, and E. Weinmüller, *New a posteriori error estimates for singular boundary value problems*, Numer. Algorithms, 40, 79-100 (2005).

[3] W. Auzinger, O. Koch, and E. Weinmüller, *Analysis of a new error estimate for collocation methods applied to singular boundary value problems*, SIAM J. Numer. Anal., 42, 2366-2386 (2005).

[4] C.J. Budd, O. Koch, and E. Weinmüller, *Computation of self-similar solution profiles for the nonlinear Schrödinger equation*, Computing, 77, 335-346 (2006).

[5] J. W. Cahn and J. E. Hilliard, *Free energy of a nonuniform system. I. Interfacial free enrgy*,J. Chem Phys., 28, 258-267 (1958).

[6] F. Dell'Isola, H. Gouin and G. Rotoli, *Nucleation and shell-like interfaces by second-gradient theory: numerical simulations*, Eur. J. Mech. B/Fluids, 15, 545-568 (1996).

[7] F. Dell'Isola, H. Gouin and P. Seppecher *Radius and surface tension of microscopic bubbles second-gradient theory*,C.R. Acad. Sci. Paris, 320 (serie IIb), 211-216 (1995).

[8] F. Gazzola, J. Serrin, and M. Tang, *Existence of ground states and free boundary problems for quasilinear elliptic operator*, Adv. Diff. Equ., 5, 1-30 (2000).

[9] M.E. Gurtin, D. Polignone, and J. Vinals, *Two-phase binary fluids and immiscible fluids described by an order parameter*, Math. Models Methods Appl. Sci., 6, 815-831 (1996).

[10] F.R. de Hoog and R. Weiss, *Difference methods for boundary value problems with a singularity of the first kind*, SIAM J. Numer. Anal., 13, 775-813 (1976).

[11] F.R. de Hoog and R. Weiss, Collocation methods for singular boundary value problems, SIAM J. Numer. Anal., 15, 198-217 (1978).

[12] F.R. de Hoog and R. Weiss, *On the boundary value problem for systems of ordinary differential equations with a singularity of the second kind*, SIAM J. Math. Anal., 11, 41-60 (1980).

[13] N. Kim, L. Consiglieri, and J.F. Rodrigues, *On non-newtonian incompressible fluids with phase transitions*, Math. Methods Appl. Sci., 29, 1523-1541 (2006) .

[14] G. Kitzhofer, O. Koch, P.M. Lima, and E. Weinmuller, *Efficient numerical solution of the density profile equation in hydrodynamics*, J. Sci. Comput., 32, 411-424 (2007).

[15] G. Kitzhofer, O. Koch, G. Pulverer, Ch. Simon, and E. Weinmüller, *The new Matlab code* `bvpsuite` *for the solution of singular implicit BVPs*, J. Numer. Anal. Indust. Appl. Math., 5, 113-134 (2010).

[16] G. Kitzhofer, *Numerical Treatment of Implicit Singular BVPs*, Ph.D. Thesis, Inst. for Anal. and Sci. Comput., Vienna Univ. of Technology, Austria, in preparation.

[17] G. Kitzhofer, O. Koch, and E. Weinmüller, *Pathfollowing for essentially singular boundary value problems with application to the complex Ginzburg–Landau equation*, BIT, 49, 217-245 (2009).

[18] O. Koch, *Asymptotically correct error estimation for collocation methods applied to singular boundary value problems*, Numer. Math., 101, 143-164 (2005).

[19] O. Koch, R. März, D. Praetorius, and E.B. Weinmüller, *Collocation methods for index-1 DAEs with a singularity of the first kind*, Math. Comp., 79, 281-304 (2010).

[20] N.B. Konyukhova, P.M. Lima, M.L. Morgado, and M.B. Soloviev, *Bubbles and droplets in nonlinear physics models: Analysis and numerical simulation of singular nonlinear boundary value problems*, Comp. Maths. Math. Phys., 48, 2018-2058 (2008).

[21] P.M. Lima, N.B. Konyukhova, N.V. Chemetov, and A.I. Sukov, *Analytical-numerical investigation of bubble-type solutions of nonlinear singular problems*, J. Comput. Appl. Math., 189, 260-273 (2006).

[22] P.M. Lima, G. Yu. Kulikov, and M. L. Morgado,*Analysis and accurate numerical approximation of singular boundary value problems with p-Laplacians in fluid mechanics*, submitted.

[23] G. Pulverer, G. Söderlind, and E. Weinmüller, *Automatic grid control in adaptive BVP solvers*, Numer. Algorithms, 56, 61-92 (2011).

[24] V. Savin, E. Valdinoci, and B. Sciunzi, *Flat level set of p-Laplace phase transitions*, Mem. Amer. Math. Soc., 182 Issue 858 (2006).

[25] B. Sciunzi and E. Valdinoci, *Mean curvature properties for p-Laplace phase transitions*,J. Eur. Math. Soc., 7, 319-359 (2005).

[26] P. Seppecher, *Moving contact lines in the Cahn-Hilliard theory*, Internat. J. Engrg. Sci., 34, 997-992 (1996).

[27] S. Wolfram, Wolfram Mathematica Documentation Centre, available at http://reference.wolfram.com/mathematica/ref/NDSolve.html?q=NDSolve&lang=en