

## Numerical regularization of Klauder effect in QLM

R. Krivec\*

Department of Theoretical Physics, J. Stefan Institute, Ljubljana, Slovenia

### ARTICLE INFO

#### Article history:

Received 7 May 2012

Received in revised form

17 July 2012

Accepted 25 July 2012

Available online 27 July 2012

#### Keywords:

Quasilinearization

Singular potentials

Spiked oscillator

Klauder effect

Numerical solution of differential equations

### ABSTRACT

The numerical solution of the spiked harmonic oscillator in quantum mechanics within the framework of the Quasilinearization Method (QLM) is investigated systematically for different degrees of singularity  $\alpha$  and different coupling constants  $\lambda$ . Several numerical algorithms consisting of different regularizations of the QLM differential equation and different nonlinear integration point distributions are studied. A simple algorithm that works uniformly across a large  $\alpha$ ,  $\lambda$  parameter space limited only by machine precision is proposed. The algorithm uses a single weak parameter which results are almost independent upon, accelerates the integration of the differential equations, and provides an automatic prescription for treating singular and regular potentials using the same methods.

© 2012 Elsevier B.V. All rights reserved.

### 1. Introduction

The spiked harmonic oscillator potential,

$$V = r^2 + \frac{\lambda}{r^\alpha} \quad (1)$$

describes interactions occurring in atomic, molecular, nuclear and particle physics, mainly in scattering problems [1,2]. For  $\alpha > 2$  it is of relevance for supersingular interactions in quantum field theory. It has been shown by Klauder [3–8] that the limit  $\lambda \rightarrow 0$  in the potential Eq. (1) does not turn off the effect of the spike. The early work by Detwiler and Klauder [3] has already shown that the matrix elements of these singular perturbations in the harmonic oscillator states diverge so that all terms in the perturbation series are infinite and the latter does not exist, as well as that the WKB method is not applicable.

Various methods combining analytical information with numerical calculation have been proposed to calculate energies and wave functions, but numerical results in the literature are usually limited to a single value of  $\alpha$ , most commonly the critical value  $\alpha = 5/2$ . An early attempt at numerical integration was Ref. [9], combining difference approximation with Richardson extrapolation. The work Ref. [10] uses the method of analytic continuation, but for  $\alpha > 2$ , when the solution acquires an essential singularity at the origin, it was necessary either to approximate the potential

or use the leading term of the solution for some  $r < R_c$ . The recent specialized method of Ref. [11] aims to provide the correct solution at the origin in addition to energies, and presents converged values for  $\alpha = 5/2$  and  $0.001 < \lambda < 20$ . The QLM (Quasilinearization Method) has also already been applied [12] to the spiked oscillator problem for a few  $\alpha$ ,  $\lambda$  values, with precision exceeding that of the literature.

Our aim in this work is to investigate the numerical limits of a standard procedure in the framework of the QLM, such that the same numerical methods could be used for regular and very singular potentials. The numerical advantage of the QLM is that its iterations are rapidly convergent for regular and singular potentials alike and self-correcting, which to some degree helps maintain numerical precision. At the present we do not concentrate on the precision of the QLM solution itself near the origin, as high-order expansions for the solution near the origin have already been found analytically using the first QLM iteration [13]. These expressions give good expectation values and could be combined with the numerical methods of the present work to give high-precision energies and wave functions. Therefore this work is devoted to a ground-up approach concentrating only on the numerical aspects, trying to construct a method that only depends “weakly” on some parameters, but nevertheless gives energies to a precision exceeding that of the literature and also as good as possible a representation of the solution near the origin, which could even suffice to calculate expectation values of interest.

The QLM pioneered by Mandelzweig [14] is a generalization, covering singular as well as regular potentials in physics, of the quadratically and often monotonically convergent scheme [15]

\* Tel.: +386 1 477 3690.

E-mail address: [rajmund.krivec@ijs.si](mailto:rajmund.krivec@ijs.si).

for solving nonlinear differential equations. It can be applied in quantum mechanics by rewriting the radial Schrödinger equation as a nonlinear Riccati equation for the logarithmic derivative  $\phi(r)$  of the wave function or a function thereof. Seemingly similar to WKB in this respect, the QLM is in fact a resummation of WKB: the  $k$ -th QLM iteration sums  $2^k$  terms [16,17] of the WKB series. In addition, for most exactly solvable potentials the first QLM iteration gives the exact energies if a quantization condition is imposed [18]. However, the Langer WKB solution [19] has been found to be an ideal initial solution for the QLM iteration [17] as it contains enough physics to assure the immediate onset of quadratic convergence of the QLM. Namely, as has been proved in Ref. [14], quadratic convergence starts at the QLM iteration where the norm of the difference of the current and previous solutions is small enough.

In our numerical application of the QLM to the one-dimensional Schrödinger equation [12,16,17,20], the QLM has been further rewritten for easier handling of nodes in  $\chi(r)$  by introducing the new function

$$\tan u(x) = -\frac{\kappa}{\phi(r)} = -\kappa \frac{\chi(r)}{\chi'(r)} \quad (2)$$

where  $x = \kappa r$ ,  $\kappa = \sqrt{E}$  with  $2m = 1$ , resulting in the nonlinear Riccati differential equation (ODE), for  $E > 0$ ,

$$u'(x) = -1 + W \sin^2 u(x) = q(x, u(x)), \quad (3)$$

where  $W(x) = 2mV(r)/\kappa^2$  (for angular momentum  $l = 0$ ). The QLM iteration equations in differential form as used in our previous works [12,20] for iterations  $k = 1, 2, \dots, M$  are

$$\begin{aligned} u'_k(x) &= q(x, u_{k-1}(x)) + (u_k(x) - u_{k-1}(x))q_u(x, u_{k-1}(x)) \\ &= Q(x, u_{k-1}(x), u_k(x)), \end{aligned} \quad (4)$$

or explicitly, leaving out some arguments for brevity,

$$\begin{aligned} Q(x, u_{k-1}, u_k) &= -1 + W \sin^2 u_{k-1} \\ &\quad + (u_k - u_{k-1})W \sin 2u_{k-1}. \end{aligned} \quad (5)$$

Here  $q_u$  denotes the functional derivative  $\partial q(x, u(x))/\partial u$ . Near the origin the solution to leading order is [13]

$$\chi(r) \approx \exp(-br^{-\beta}) \quad (6)$$

or

$$u(x) \approx -\frac{\kappa}{\sqrt{\lambda}} \left(\frac{x}{\kappa}\right)^{\alpha/2} \quad (7)$$

where  $\beta = \alpha/2 - 1$ ,  $b\beta = \sqrt{\lambda}$ .

In the previous work [12] it was found that stable numerical convergence is obtained by using a simple explicit solver (Runge–Kutta) on a nonuniform point distribution where the point density is roughly inversely proportional to the size of the derivative of the function  $\chi(r)$ , which is nonanalytic at the origin, on the interval left of the inflection point,  $\chi''(r) = 0$ . (This ensured that the resulting point separation monotonically increases with  $r$ .) It was not sufficient in general to use a density inversely proportional to the derivative of the analytic function  $u(\kappa r)$ . This point distribution was more nonlinear than the one used via variable substitution in Ref. [21],  $1 + Kr = e^{ky}$ . However, only a few  $\alpha, \lambda$  values were selected and the applicability of the numerical method to other values was not investigated in detail in the work [12].

The goal of this work is (i) to extend the  $\alpha, \lambda$  parameter space where the standard QLM code based on the Riccati ODE can be applied, and (ii) to try to make integration parameter selection automatic in this space, implying that the parameters should vary slowly with  $\alpha, \lambda$ . We use the nonuniform point distribution of

Ref. [12] as a starting point. This will keep a unified framework for the QLM iteration. For example, in our case a variable substitution numerically sensible near the origin may not be sensible at large  $x$ . We concentrate on a possibly very nonlinear point distribution instead, and investigate the limits of its numerical implementation imposed by the number representation in the computer.

The Klauder effect manifests itself in the fact that in Eqs. (4)–(5),  $q(0, u(0)) = Q(0, u_{k-1}(0), u_k(0)) = 0$ , in contrast to regular potentials where  $q(0, u(0)) = Q(0, u_{k-1}(0), u_k(0)) = -1$ . Numerically this is reflected in the need for the terms of  $Q(x, u_{k-1}, u_k)$ , Eq. (5), to cancel out near the origin. The problem lies not with the very large values of  $W(x)$ , which are evaluated numerically in a stable manner, but with the following three issues in the evaluation of  $Q(x, u_{k-1}, u_k)$ .

First, the numerical initial solution  $u_0(x)$  for the QLM iteration may simply be zero either identically or on a finite interval near the origin, for various reasons including lower precision attained by available tools. Indeed, in the QLM the numerical precision of the initial approximation should not matter, and should not be expected. (For example, within a 128-bit QLM iteration scheme we may be using a Langer WKB solution calculated in 64-bit arithmetic using Airy functions by a 64-bit numerical library [22].) On the interval where  $u_0(x) = 0$ , we get numerically  $Q(x, u_0, u_1) = -1$ , which effectively removes the singularity of  $V$ .

Second, the higher iterations  $u_k(x)$  suffer from integration errors which depend on the error of  $Q(x, u_0, u_1)$  and on the way this error propagates to  $Q(x, u_{k-1}, u_k)$  for  $k > 1$ . (This would especially be the case if an adaptive solver were used and, as is sensible, the absolute error of the solution were checked near the origin instead of its relative error.) The value of  $u_k(x)$  returned by the solver, whose correct value near the origin is very small (if point density is large as anticipated), may consequently vary by orders of magnitude from the true value, and, despite still being “small”, prevent correct cancellations in  $Q(x, u_{k-1}, u_k)$ . We address this by regularizing the values of  $u_{k-1}(x)$  before they enter the  $Q(x, u_{k-1}, u_k)$  calculation, or the value of the resulting  $Q(x, u_{k-1}, u_k)$ , or both.

Third, the problem has two scales. The quasiclassical region of  $x$  where the matching point is normally set depends predominantly on the harmonic part of  $V$  and varies relatively slowly with  $\alpha$  and  $\lambda$ , compared to the inflection point of Eq. (6),  $r_c = (\sqrt{\lambda}/(\beta + 1))^{1/\beta}$ , which is of the order of  $10^{-8}$  at  $\alpha = 2.5, \lambda = 0.0001$  and  $10^6$  at  $\alpha = 2.5, \lambda = 1000$ . This could lead to nonmonotonic point separations and numerical underflow of point separation near the origin.

In general, while the QLM is self-correcting, numerical noise has similar effects on convergence as a bad initial solution: it may delay, or, in the case of singular potentials, preclude the onset of quadratic convergence.

## 2. Acceleration and regularization prescriptions

We present an improved nonlinear point distribution which provides the main numerical acceleration, and different regularization prescriptions that control the numerical noise near the origin. Combinations of point distributions and regularization prescriptions are shown schematically in Table 1.

Nonuniform point distributions are only applied on the LHS of the matching point ( $0 \leq x \leq x_m$ ), while on the RHS we use a uniform distribution. The different types of nonuniform point distributions  $\{x_i\}$  in Table 1 are based on equidistant sets  $\{z_i\}$  as follows:

- A (old):  $x_i = \exp(-bz_i^{-\beta'})$  from  $x = 0$  to the inflection point  $x_c = \kappa r_c$  of the distribution; this makes the point density behave like the derivative of the solution  $\chi$ . Points are defined

**Table 1**

Schematic representation of regularization algorithms, labelled by point distribution type (A, B, C) and regularization type (0, Q, U, UQ).  $\{z_i\}$  is a set of equidistant values.

| Regularization:                          | Point distribution, $\{x_i\}$              |  |                      |
|--|--|--|----------------------|
|  | $\exp(-bz_i^{-\beta'})$ plus power law (A) | $a \exp(-bz_i^{-\beta}) x_N = x_m$ (B) | $z_i^{\alpha/2}$ (C) |
| $u$ and $Q$ leading terms, $i < I_x$ (0) | A0   |  |                      |
| $Q$ replacement ( $q$ )                  | Aq   |  |                      |
| $u$ minorization ( $U$ )                 | AU   | BU                                     |                      |
| $u$ and $Q$ minorization ( $UQ$ )        |  | BUQ                                    | CUQ                  |

left-to-right by setting the first separation at  $x = 0$  to be  $x_1 - x_0 = 10^{-\sigma x}$ ; if then  $x_c > x_m$  we bring it into  $[0, x_m]$  by defining a different  $\beta' \neq \beta$ ; the point distribution is then padded with a power-law distribution on the right if the new  $x'_c < x_m$ . This works only for a small region of  $\alpha, \lambda$ .

- B: a point distribution defined right-to-left scaled by a factor  $a$  in the  $x$  direction so that the resulting distribution inflection point always coincides with  $x_m$ :  $x_i = a \exp(-bz_i^{-\beta})$ ,  $a = \exp(\alpha/(\alpha - 2))$ . Points near the origin may get very dense; the set of points that differ by less than  $10^{-\sigma x}$  (contiguous by virtue of monotonicity) is replaced by a linear distribution of the same number of points. This may reduce the final minimum separation below  $10^{-\sigma x}$ , but it does so very weakly, as  $O(N^{-1})$ , compared with the highly nonlinear separation dependence of the original distribution.
- C: power-law point distribution  $x_i = z_i^{\alpha/2}$  (shaped like  $u(x)$ ).

The different types of regularizations in Table 1 are as follows:

- 0 (old):  $u_{k-1}(x)$ ,  $u_k(x)$  and  $Q(x, u_{k-1}, u_k)$  are replaced by the leading order explicit solutions on the first  $I_x$  points near the origin.  $I_x$  is fixed during parameter optimization (cf. Section 3) so that the regularization interval  $[0, x_{I_x}]$  diminishes as  $N$  increases.
- $q$  (old): replacement of  $Q(x, u_{k-1}, u_k)$  by the leading term of the explicit expression for those  $x_i$  where this leading term is smaller than the current numerical solution. This avoids an explicit  $x$  cutoff and  $u'(x)$  discontinuity but still fixes the form of  $u(x)$  below the crossing point of the numerical solution and the leading term.
- $U$ : minorization of  $u_{k-1}(x)$  and  $u_k(x)$  before the  $Q(x, u_{k-1}, u_k)$  calculation, using a multiple of the leading term of the explicit solution. This does not enforce the  $u_k(x)$  values themselves, just limits their numerical noise.
- $UQ$ : minorization of  $Q(x, u_{k-1}, u_k)$  in addition to  $u_{k-1}(x)$ ,  $u_k(x)$ .

The minorization function for  $u(x)$ ,

$$f_m(x) = F_m \left( -\frac{\kappa}{\sqrt{\lambda}} \left( \frac{x}{\kappa} \right)^{\alpha/2} \right), \quad (8)$$

is just the leading term of  $u(x)$  multiplied by an arbitrary parameter  $F_m$ . If the numerical solution  $u_k(x)$  is smaller than  $f_m(x)$  (larger in absolute value), it is replaced by  $f_m(x)$ , and if  $u_k(x) > 0$ , it is replaced by 0. If  $Q(x, u_{k-1}, u_k)$  is minorized, it is likewise confined between  $f'_m(x)$  and 0. Minorization has no effect if the numerical solution lies between  $f_m$  and 0.  $f_m$  is applied at all  $x$ , even though it is not generally true that  $f_m(x)$  is smaller than the true solution. It certainly is true for large enough  $x$ , which is ensured by omitting the tangent function in Eq. (2). The ambiguity in the crossing point of the numerical solution and of  $f_m$  therefore consists of the unknown effects of omitting the tangent function and additional terms in  $f_m$  and of the effect of  $F_m$ . The effect of this ambiguity on the results is resolved by performing computations with several fixed values of the “external” parameter  $F_m$ . The advantage here is that there is no explicit cutoff parameter so the procedure is simple and automatic; also, additional computations for different  $F_m$  serve as independent verifications.

### 3. Description of calculation

The approximate  $E$  value of the ground or an excited state for a given  $\alpha, \lambda$  pair is first determined by a low-precision tabulation of the difference of the RHS and LHS solutions at the matching point  $D = u_{M,\text{LHS}}(x_m) - u_{M,\text{RHS}}(x_m)$  as a function of  $\kappa$ .

The high-precision calculations for a given  $\alpha, \lambda$  pair proceed as follows. Eq. (4) is solved in two phases for different “external” parameters like  $F_m$  and  $\sigma_x$  and the selected algorithm from Table 1. The first phase is the parameter optimization phase in which  $\kappa$  is fixed close to the true  $\kappa$ , and the “internal” parameters  $N, M$ , and  $x_u$  are given some initial values  $N_0, M_0$  and  $x_{u0}$  and then gradually increased in subsequent passes in the order of the effect their previous change has had on  $D$ , until  $|D|$  is small enough. In each optimization pass the entire QLM iteration is repeated, starting with the (possibly lower precision) initial solution of the Langer WKB type. (The energy of the WKB solution is optionally recalculated in each pass independently of  $\kappa$ .) The second phase is the search for the zero of  $D$  as a function of  $\kappa$ , using the optimized “internal” parameters and the fixed “external” parameters; the zero gives  $E$ .

Integration in this work has been performed using the fourth-order explicit Runge–Kutta method in order to preserve full control over integration. In particular, this eliminates local error estimation and subsequent step and order adjustments, whose effects could not be separated from the effects of different point densities and regularizations. To integrate the  $k$ -th QLM iteration  $u_k(x)$ , the Runge–Kutta method however requires the values of the previous function  $u_{k-1}(x)$  at midpoints of the integration steps. These values are obtained by a five-point Lagrange interpolation [23] whose estimated error is one order in  $h$  smaller [24] than the estimated solver error. Testing of an alternative method avoiding the interpolation phase is in progress.

For the level of precision of  $E$  reported in this work a 128-bit floating-point representation, nowadays readily available in efficient compiler implementations (as opposed to unlimited precision libraries), is required.

### 4. Results and discussion

Typical behavior of  $u_k(x)$  for successive QLM iterations  $k = 1, 2, \dots, M$  near the origin for new regularization algorithms is illustrated in Figs. 1–3 for a large  $\lambda$  where it is more pronounced. Incidentally in the  $x$  intervals displayed in these figures the numerical initial WKB approximation is identically zero.  $E$  values agree to 21 significant digits despite variations in  $u_k(x)$  near the origin if  $Q(x, u_{k-1}, u_k)$  is not minorized, in which case  $Q(x, u_{k-1}, u_k)$  starts out with the wrong value of  $-1$ . The bottom graphs of Figs. 1–2 in logarithmic scale show the numerical noise below the crossing point of  $u_{k-1}(x)$  and  $f_m$ , which the algorithm BUQ in Fig. 3 practically eliminates.

Tables 2–10 present  $E$  values calculated with old and new regularization algorithms, rounded to the number of digits requested from the parameter optimization algorithm. Usually at least one more digit than displayed is correct in the results, but occasionally the last digit of lower-precision result is rounded off incorrectly,

**Table 2**  
 E for selected combinations of “external” parameters  $F_m$  and  $\sigma_x$ , point distributions, and regularization prescriptions (“R.A.”),  $\alpha = 2.5$ ,  $\lambda = 0.0001$ .  $10^{-\sigma_x}$  is the smallest point separation parameter near the origin, and its use depends slightly on the algorithm specified in column “R.A.”  $F_m$  is the size parameter for the minorization function.  $x_m$  and  $x_u$  are the matching point and the upper interval boundary, respectively;  $x_m$  is constant, given by the condition  $u(x_m) \approx -\pi/2$ , while  $x_u$  is an “internal” parameter increased during parameter optimization.  $P$  is the number of significant digits after the comma; the adjacent column (s.) specifies the solver used (RK: Runge–Kutta, A: adaptive variable order/variable step solver [25]).  $M$  is the maximum QLM iteration (either an “internal” parameter or fixed);  $D = u_{M,RHS}(x_m) - u_{M,LHS}(x_m)$ .  $N$  is the number of points on  $[0, x_m]$  and  $[x_m, x_u]$ ; for Runge–Kutta  $P \approx O(N^{-4})$ . For illustration, the  $M = 3$  result (row 1) has converged to 5 digits less than the 4-th QLM iteration; large  $N$  is an artifact of a conservative  $M$ -increasing algorithm. Results for  $\sigma_x = 32$  and 64 agree to all digits; in several cases  $M$  up to 12 has been tested and results agree with  $M = 4$ .

| $\sigma_x$ | $F_m$    | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$        | $E$                        |
|------------|----------|------|-------|-------|-----|----|-----|------------|----------------------------|
| 64         | 1.200    | BU   | 1.5   | 12.62 | 20  | RK | 3   | 1 490 769  | 3.00040789861816370549     |
| 64         | 1.000    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 899 178    | 3.00040789861816316882     |
| 64         | 1.001    | BUQ  | 1.5   | 12.87 | 20  | RK | 8   | 897 296    |                            |
| 64         | 1.010    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 899 178    |                            |
| 64         | 1.050    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 894 069    |                            |
| 64         | 1.100    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 898 783    |                            |
| 64         | 1.200    | BU   | 1.5   | 12.87 | 20  | RK | 4   | 899 756    |                            |
| 64         | 1.300    | BUQ  | 1.5   | 12.87 | 20  | RK | 8   | 903 483    |                            |
| 64         | 1.500    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 899 178    |                            |
| 64         | 2.000    | BU   | 1.5   | 12.87 | 20  | RK | 8   | 899 427    |                            |
| 64         | 8.000    | BUQ  | 1.5   | 12.87 | 20  | RK | 8   | 905 966    |                            |
| 64         | 100.000  | BU   | 1.5   | 12.87 | 20  | RK | 8   | 895 609    |                            |
| 64         | 1000.000 | BU   | 1.5   | 39.00 | 20  | RK | 8   | 951 646    |                            |
| 32         | 1.300    | BU   | 1.5   | 12.87 | 20  | A  | 8   | 209 083    | 3.00040789861816316882     |
| 64         | 1.300    | BU   | 1.5   | 12.87 | 20  | A  | 8   | 209 083    |                            |
| 64         | 1.300    | BU   | 1.5   | 12.87 | 20  | A  | 8   | 2 090 956  |                            |
| 32         | 1.001    | BUQ  | 1.5   | 14.21 | 24  | RK | 8   | 10 505 710 | 3.000407898618163168821759 |
| 32         | 1.300    | BU   | 1.5   | 14.21 | 24  | RK | 6   | 10 527 752 |                            |
| 32         | 1.300    | BUQ  | 1.5   | 14.21 | 24  | RK | 8   | 10 578 156 |                            |

**Table 3**  
 As Table 2, but  $\alpha = 2.5$ ,  $\lambda = 1$ . The old method AU exhibits a slight instability.

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$        | $E$                            |
|------------|-------|------|-------|-------|-----|----|-----|------------|--------------------------------|
| 32         | 0.000 | AU   | 1.8   | 15.83 | 20  | RK | 8   | 282 137    | 4.31731168924736736599         |
| 32         | 0.000 | AU   | 2.8   | 15.83 | 20  | RK | 8   | 282 178    | 4.31731168924736736598         |
| 64         | 1.200 | BU   | 3.0   | 16.08 | 20  | RK | 4   | 233 691    | 4.31731168924736736599         |
| 64         | 1.300 | BU   | 3.0   | 18.11 | 27  | RK | 8   | 13 470 653 | 4.317311689247367365989407606  |
| 64         | 1.300 | BU   | 3.0   | 18.48 | 28  | RK | 8   | 23 220 709 | 4.3173116892473673659894076064 |

**Table 4**  
 As Table 2, but  $\alpha = 2.5$ ,  $\lambda = 1000$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$  | $P$ | s. | $M$ | $N$        | $E$                           |
|------------|-------|------|-------|--------|-----|----|-----|------------|-------------------------------|
| 64         | 1.200 | BU   | 30.0  | 112.62 | 19  | RK | 4   | 469 118    | 44.9554847880956299040        |
| 32         | 1.300 | BUQ  | 30.0  | 117.17 | 23  | RK | 8   | 5 175 722  | 44.95548478809562990399132    |
| 64         | 1.300 | BU   | 30.0  | 121.90 | 26  | RK | 8   | 25 991 492 | 44.95548478809562990399132464 |

**Table 5**  
 As Table 2, but  $\alpha = 3.5$ ,  $\lambda = 0.0001$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$        | $E$                           |
|------------|-------|------|-------|-------|-----|----|-----|------------|-------------------------------|
| 64         | 1.200 | BU   | 2.0   | 13.66 | 20  | RK | 3   | 557 943    | 3.00786463633627324667        |
| 64         | 1.200 | BU   | 2.0   | 12.87 | 20  | RK | 4   | 316 722    | 3.00786463633627296200        |
| 32         | 1.300 | BUQ  | 2.0   | 14.21 | 24  | RK | 8   | 2 810 891  | 3.007864636336272961998877    |
| 64         | 1.300 | BU   | 2.0   | 14.78 | 27  | RK | 8   | 18 512 572 | 3.007864636336272961998877131 |

**Table 6**  
 As Table 2, but  $\alpha = 3.5$ ,  $\lambda = 1$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$        | $E$                            |
|------------|-------|------|-------|-------|-----|----|-----|------------|--------------------------------|
| 64         | 1.200 | BU   | 3.0   | 16.08 | 20  | RK | 4   | 233 667    | 4.44294789891022795321         |
| 32         | 1.300 | BUQ  | 3.0   | 17.41 | 24  | RK | 8   | 2 063 194  | 4.442947898910227953212778     |
| 64         | 1.300 | BU   | 3.0   | 18.85 | 28  | RK | 8   | 23 682 661 | 4.4429478989102279532127782565 |

**Table 7**  
 As Table 2, but  $\alpha = 3.5$ ,  $\lambda = 1000$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$  | $P$ | s. | $M$ | $N$        | $E$                           |
|------------|-------|------|-------|--------|-----|----|-----|------------|-------------------------------|
| 64         | 1.200 | BU   | 20.0  | 143.40 | 19  | RK | 3   | 2 187 263  | 26.1088452422553116909        |
| 64         | 1.200 | BU   | 20.0  | 67.57  | 19  | RK | 4   | 360 860    | 26.1088452422552341817        |
| 32         | 1.300 | BUQ  | 20.0  | 70.30  | 23  | RK | 8   | 3 062 558  | 26.10884524225523418166949    |
| 64         | 1.300 | BU   | 20.0  | 71.71  | 26  | RK | 8   | 19 601 431 | 26.10884524225523418166948949 |

**Table 8**

As Table 2, but  $\alpha = 7.0, \lambda = 0.0001$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$       | $E$                        |
|------------|-------|------|-------|-------|-----|----|-----|-----------|----------------------------|
| 64         | 1.200 | BU   | 2.0   | 13.66 | 20  | RK | 4   | 671 509   | 3.24626398251223449657     |
| 32         | 1.300 | BUQ  | 2.0   | 14.78 | 24  | RK | 12  | 5 929 228 | 3.246263982512234496569367 |

**Table 9**

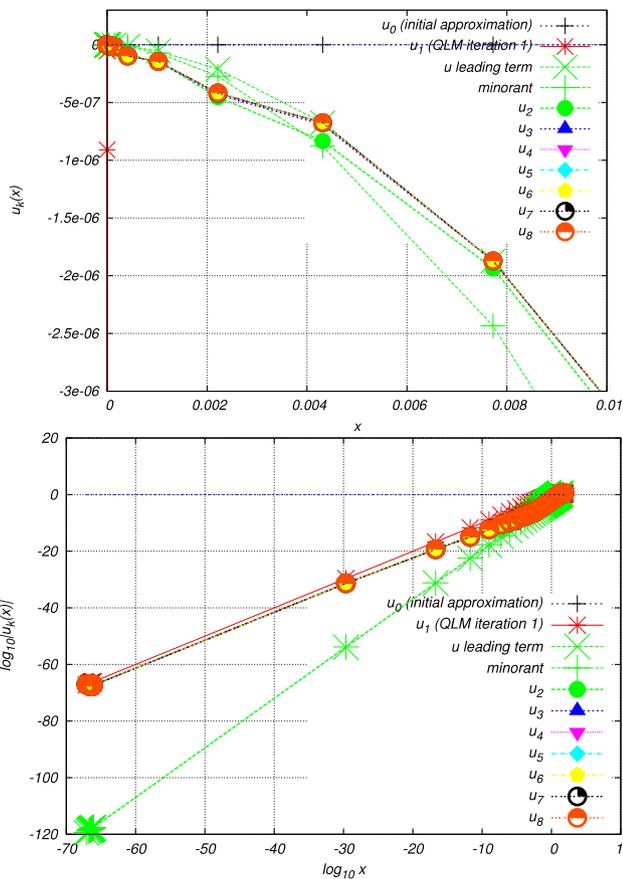
As Table 2, but  $\alpha = 7.0, \lambda = 1$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$       | $E$                        |
|------------|-------|------|-------|-------|-----|----|-----|-----------|----------------------------|
| 64         | 1.200 | BU   | 4.0   | 16.89 | 20  | RK | 4   | 576 303   | 4.72801534446030414488     |
| 32         | 1.300 | BUQ  | 4.0   | 18.28 | 24  | RK | 8   | 6 615 144 | 4.728015344460304144887804 |

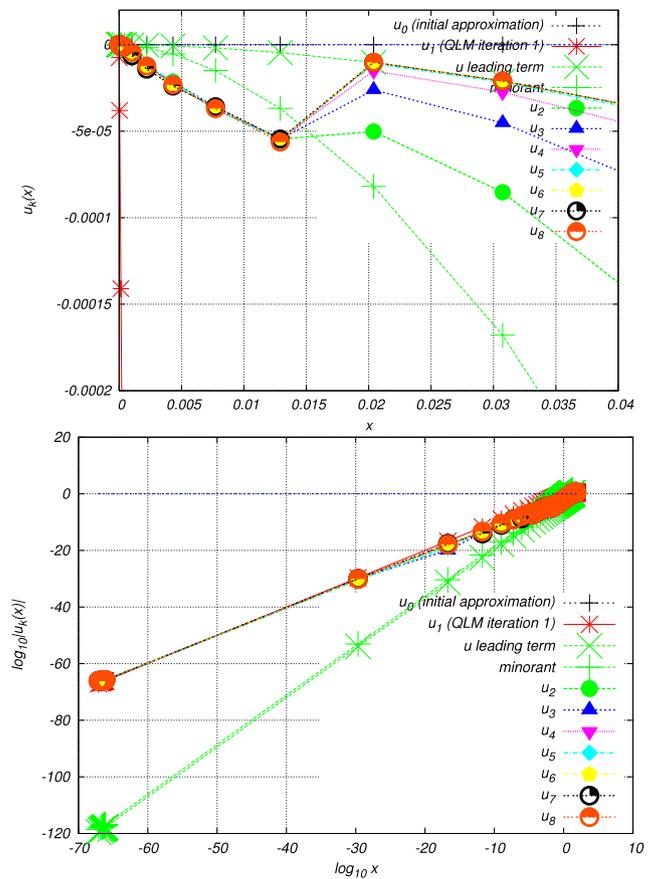
**Table 10**

As Table 2, but  $\alpha = 7.0, \lambda = 1000$ .

| $\sigma_x$ | $F_m$ | R.A. | $x_m$ | $x_u$ | $P$ | s. | $M$ | $N$       | $E$                        |
|------------|-------|------|-------|-------|-----|----|-----|-----------|----------------------------|
| 64         | 1.200 | BU   | 8.0   | 33.78 | 19  | RK | 4   | 609 853   | 10.8966840384220026627     |
| 32         | 1.300 | BUQ  | 8.0   | 35.85 | 23  | RK | 8   | 6 863 006 | 10.89668403842200266271998 |



**Fig. 1.**  $u_k(x_i), k = 0, \dots, 8$  for  $\alpha = 3.5, \lambda = 1000$ , algorithm BU ( $u_{k-1}$  and  $u_k$  regularized,  $Q(x_i, u_{k-1}, u_k)$  not regularized),  $F_m = 1.3$ ; parameters optimized for 20 significant digits, converged  $E = 26.1088452422552341817$ . QLM iterations  $k \geq 4$  are indistinguishable in the graphs;  $u_k$  stabilize for  $x_i > 0.005$ ; here  $x_m = 20, x_u = 67.57$ . Top: linear scale, bottom:  $x_i, |u_k(x_i)|$  in logarithmic scale.  $u_k$  values are calculated by the ODE solver from regularized  $u_{k-1}$  and are not yet regularized themselves, therefore they can undershoot the minorant. The initial Langer WKB solution  $u_0$  is zero for  $x_i < 0.12$  due to lower computational precision. “ $u$  leading term” denotes  $f_m(x_i)/F_m$ ; “minorant” denotes  $f_m(x_i)$ . Displayed interval contains about  $5 \times 10^4$  integration points of the  $N/2 = 360\,860$  points on  $[0, x_m]$ , drawn with  $i$  running in steps of  $N/200$  except for  $i \leq 11$  where  $i$  step is 1.

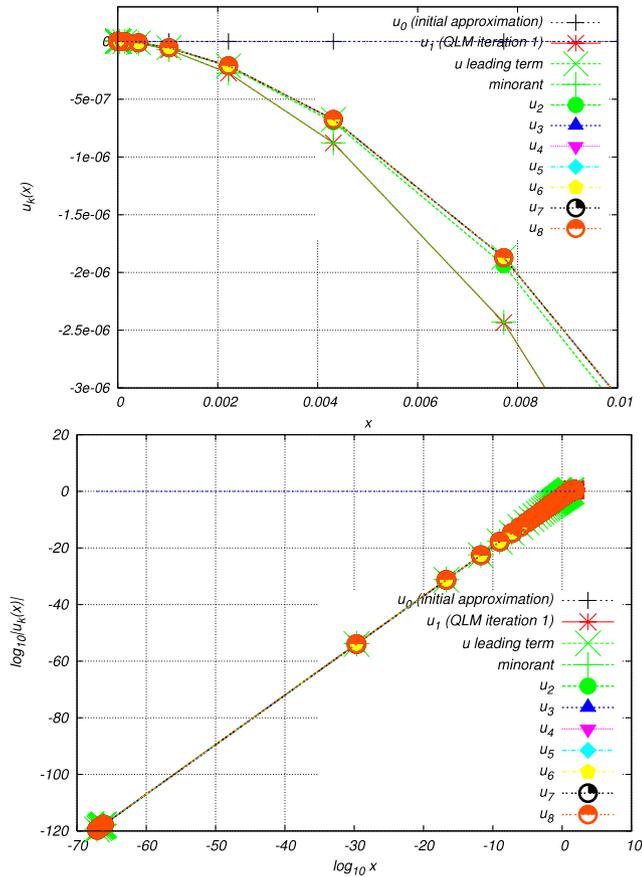


**Fig. 2.** As in Fig. 1, but  $F_m = 8$ , an unrealistically large value. (Note that scale is different than in Fig. 1.) Nevertheless the  $E$  value agrees to all digits with the value quoted in Fig. 1. Wrong  $u_k$  values with slope  $\approx -1$  persist on a larger interval, and an additional QLM iteration is required for indistinguishability of  $u_k$ .

which is entirely due to the error estimation algorithm and not to the intrinsic precision of the method. The column  $P$  refers to the number of converged digits behind the comma, not the number of significant digits. Up to 29 significant digits precision is displayed;

this is rather close to the limit of the 128-bit arithmetic but easily reached, as illustrated by a few values (Tables 3 and 6); 28 digits are displayed in Tables 4, 5 and 7; the somewhat incomplete set of results with more than 28 digits was only limited by the available computer time.

Precision of the order of 20 digits in  $E$  is achieved with only four QLM iterations, as already indicated on the limited number of cases in the previous work [12]; this is even less than in the QLM calculations of regular or Coulomb-like potentials using equidistant integration points in Refs. [17, 18].  $M = 3$  is always too



**Fig. 3.** As in Fig. 1, but algorithm BUQ ( $u_{k-1}$ ,  $u_k$  and  $Q(x, u_{k-1}, u_k)$  regularized). The  $E$  value is again identical to that of Fig. 1, and all QLM iterations  $u_k(x)$  are indistinguishable at this scale (while QLM iterations of Fig. 1 agree only for  $x > 0.005$ ).

small (Tables 2, 5 and 7). The Langer WKB initial approximation energy agrees with the exact  $E$  to only a few digits, therefore each QLM iteration yields 4–5 digits in  $E$  for the spiked oscillator potential. In view of this we usually fix  $M$  to typical values 4, 8, 12, ... and perform separate computations using different  $M$ , thus eliminating one convergence parameter.

For all Tables 2–10 control runs with one of the “internal” parameters larger than usual were made. About half of the typical scope of control runs in  $F_m$  and in the two methods BU and BUQ is only presented in Table 2. The latter control runs demonstrate the resilience of  $E$  to residual errors of  $u_k(x)$  near the origin. Some tables display control runs in  $M$ .

4.1. Comparison of algorithms

Older algorithms used in the previous work [12] employing the inferior point distribution A are slow and unpredictable. Algorithm A0 converges fast in some cases (small  $\alpha$  or  $\lambda$ ), and suffers from arbitrariness of the parameter  $I_x$ . Algorithm Aq converges uniformly over  $\alpha, \lambda$  but extremely slowly (needs of the order of  $10^5$  points to reach a few digit precision). Algorithm AU is only superior to Aq for  $\alpha = 2.5, \lambda = 0.0001$ ; for  $\alpha = 2.5, \lambda = 1$  AU is only a little better than Aq and fails at other  $\alpha, \lambda$ .

New algorithms BUQ and BU using the improved point distribution B require only about  $10^3$  points for 12 digit precision, and converge uniformly over  $\alpha, \lambda$  (and require about half as many points as algorithm A0 where the latter converges at all). Algorithm BUQ does not provide acceleration of (or better  $E$  than) BU, but provides much better numerical solutions  $u_k(x)$

at small  $x$  (on the scale defined by  $x_m$ ) (see Figs. 1 and 3). For comparison, the numerical method used in [9] (approximating the second derivative with centered differences followed by Richardson extrapolation) required about 80,000 points for 6 significant digits.

Like the old algorithm AU in Ref. [12], the results by new algorithms BUQ and BU were once again checked against the algorithm CUQ, which uses a simpler power-law point distribution modeled after the  $u(x)$  leading term near the origin. As expected CUQ is of a small advantage only at  $\lambda \gg 1$  and converges much slower than BUQ at  $\lambda \ll 1$ . At  $\alpha = 2.5, \lambda = 0.0001$  CUQ requires  $N = 165\,031$  and 1600 s for 8 significant digits in  $E$  compared with BU or BUQ, which require  $N = 510$  and 5 s.

As anticipated, the choice of  $F_m$  is rather arbitrary; values in even as unrealistic an interval as  $[0.00001, 100]$  have been checked to give the same  $E$  values (Tables 2–10), except that with the algorithm BU, the QLM iterations or the direct solution of  $u(x)$  deviate appreciably from the correct solution, the more so the more  $F_m$  deviates from 1 (Figs. 1–2), while with the algorithm BUQ they are almost indistinguishable (Fig. 3). However, despite its arbitrariness  $F_m$  becomes important when large enough; it should be borne in mind that without minorization (or equivalently by setting  $F_m = \infty$ ), the integration does not succeed at all. The “external” parameter  $\sigma_x$  can also be safely set to a large number; e.g. twice the number of digits in machine precision.

4.2. Summary

New algorithms BU and BUQ used with the Runge–Kutta method work uniformly well for  $E$  for all checked  $\alpha, \lambda$  combinations, and BUQ in addition gives a surprisingly good solution  $u(x)$  near origin. The parameter  $F_m$  can be conveniently put equal to, say, 1.3, as its precise value is unimportant for  $E$  values, and likewise  $\sigma_x$  can be set to e.g. 64. In this sense the present results can be obtained without a free parameter.

Algorithm BUQ provides both acceleration of the ODE solution (reduction of the number of integration points for given precision by moving them where they are needed) and better values of the solution (a reduction of error by many orders of magnitude) near the origin (in terms of the scale of the integration intervals  $[0, x_m]$  and  $[x_m, x_u]$ ). This enabled the automation of the computations by setting up tables of starting values of “internal” convergence parameters  $M_0, N_0$  and  $x_{u0}$  (a small multiple of  $x_m$ ).  $M_0$  and  $N_0$  vary very slowly across the  $\alpha, \lambda$  space and do not depend on the required precision. The parameters  $x_m$  and  $x_u$  vary in the  $\alpha, \lambda$  space but  $x_m$  is determined by the overall scale of  $V$  via the Langer WKB solution such that  $u_{\text{WKB}}(x_m) \approx -\pi/2$ . Namely, this value is attained by  $u(x)$  at the upper bound of the left-to-right integration stability interval where the Jacobian

$$J_k(x) = \frac{\partial Q(x, u_{k-1}(x), u_k(x))}{\partial u_k} = W \sin 2u_{k-1}(x) \tag{9}$$

is negative.

The proposed regularization is stable and efficient for Runge–Kutta but does not appear suitable for adaptive solvers, usually of the combined Adams/BDF type, which operate with prescribed accuracy. Standard adaptive solvers [25] usually fail well below  $P = 20$ , even in 128-bit arithmetic. The precise reason has not been investigated, but since they typically try to subdivide point separation indefinitely, we assume this is because internal intermediate values suffer from the type of numerical noise treated in our regularization algorithms. One way to approach this would be to specify smaller precision requirements near the origin, but that would introduce additional parameters in the regularization; this may be the subject of another work. Inserting regularization

code directly into adaptive solvers (unlike the Runge–Kutta solver where no error estimates are performed and regularization can be done *a posteriori*) may not make sense as error estimates [24] are based on approximations to the Jacobian equation (9), which is singular at the origin,  $J_k(x) \approx x^{-\alpha/2}$ , in addition to being subject to numerical noise.

Computational times using an explicit fourth-order Runge–Kutta method in quadruple (128-bit) precision are of the order of 10 min for 20 significant digits; no special attempt to optimize the program, e.g. by pre-tabulating  $V$ , etc., has been made. The complete times including “internal” parameter optimization (primarily in  $N$ ) are of the order of 1–2 h, as we always started from conservative parameter values in order to perform redundant checking and to determine the minimal required parameter values. However, as the values of  $N$  and  $M$  in Tables 2–10 vary rather slowly they can be safely interpolated and extrapolated in  $\alpha$ ,  $\lambda$  space, performing parameter optimization only on some  $\alpha$  values and on  $\lambda$  values separated by several orders of magnitude.

#### 4.3. Conclusion

The proposed numerical approach puts the numerical difficulties of integrating the QLM equations for the spiked harmonic oscillator under the control of a single “weak” regularization parameter  $F_m$  that has negligible influence on  $E$  on a very large interval, but is critical for the approach to work in the sense that removing the corresponding regularization (or setting this parameter to infinity) causes the integration to fail. The precise form of  $f_m$ , i.e. the number of expansion terms [13] included may be largely irrelevant for the determination of  $E$ . Singularities other than power-law type may be treatable in the same way. Selection of  $f_m$  is facilitated by our choice of solution representation, as merely omitting the tangent function usually makes it a minorant on the entire interval. Our results cover a rectangular region of the  $\alpha$ ,  $\lambda$  space, which includes  $\alpha$ ,  $\lambda$  combinations published previously.

In addition to absolutely stable  $E$  values, we obtained an unusually stable and smooth form of the solution near the origin, which may turn out to be sufficiently accurate to calculate expectation values. We plan to test the need for improvements of the wave function near the origin by combining the present numerical approach with the expansions evaluated analytically from the 1st QLM iteration [13].

The results were made possible by a simple explicit formula for the nonuniform integration point distribution which depends on the coefficients of the spike,  $\lambda/r^\alpha$ . This point distribution bridges the small scale of the spike and the scale of the harmonic term. A simple Runge–Kutta solver was used in this work. The integration parameters like the number of integration points vary very slowly across  $\alpha$ ,  $\lambda$ , which enabled a practically automatic calculation of energies up to the level of just a few digits below the machine precision. Moreover, the relaxation of  $F_m$  towards large values can be used as an estimate of the influence of the solution errors in a small region near origin on the results. The required number of QLM iterations for 20 digit precision (four) is smaller than in the calculations of regular potentials using equidistant points [17,18].

#### Acknowledgment

I greatly appreciate discussions with V.B. Mandelzweig.

#### References

- [1] W. Bühring, J. Math. Phys. 15 (1974) 1451.
- [2] W.M. Frank, D.J. Land, R.M. Spector, Rev. Modern Phys. 43 (1971) 36.
- [3] L.C. Detwiler, J.R. Klauder, Phys. Rev. D 11 (1975) 1436.
- [4] J.R. Klauder, Acta Phys. Austriaca (Suppl. 11) (1973) 341.
- [5] J.R. Klauder, Phys. Lett. B 47 (1973) 523.
- [6] J.R. Klauder, Science 199 (1978) 735.
- [7] H. Ezawa, J.R. Klauder, L.A. Shepp, J. Math. Phys. 16 (1975) 783.
- [8] B. Simon, J. Funct. Anal. 14 (1973) 295.
- [9] V.C. Aguilera-Navarro, G.A. Estévez, R. Guardiola, J. Math. Phys. 31 (1990) 99.
- [10] E. Buendía, F.J. Gálvez, A. Puertas, J. Phys. A: Math. Gen. 28 (1995) 6731.
- [11] F.J. Gómez, J. Sesma, J. Phys. A 43 (2010) 385302.
- [12] R. Krivec, V.B. Mandelzweig, Comput. Phys. Comm. 179 (2008) 865.
- [13] E.Z. Liverts, V.B. Mandelzweig, Ann. Physics 322 (2007) 2211.
- [14] V.B. Mandelzweig, J. Math. Phys. 40 (1999) 6266.
- [15] R.E. Bellman, R.E. Kalaba, Quasilinearization and Nonlinear Boundary-Value Problems, Elsevier Publishing Company, New York, 1965.
- [16] R. Krivec, V.B. Mandelzweig, F. Tabakin, Few-Body Syst. 34 (2004) 57.
- [17] R. Krivec, V.B. Mandelzweig, Comput. Phys. Comm. 174 (2006) 119.
- [18] R. Krivec, V.B. Mandelzweig, Phys. Lett. A 337 (2005) 354.
- [19] R.E. Langer, Phys. Rev. 51 (1937) 669.
- [20] R. Krivec, V.B. Mandelzweig, Comput. Phys. Comm. 152 (2003) 165.
- [21] J.P. Killingbeck, G. Jolicard, A. Grosjean, J. Phys. A: Math. Gen. 34 (2001) L367.
- [22] NAG Numerical Library, The Numerical Algorithms Group Ltd., Oxford, 2006.
- [23] M. Abramovitz, I. Stegun, Handbook of Mathematical Functions, Dover, New York, 1972, p. 879.
- [24] E. Isaacson, H.B. Keller, Analysis of Numerical Methods, J. Wiley & Sons, New York, 1966.
- [25] <http://www.netlib.org/odepack/index.html>.