RECENT IMPROVEMENTS TO A VERSION OF THE WILSON-DEVINNEY PROGRAM

JOSEF KALLRATH

BASF-AG, ZX/ZC-C13, 67056 Ludwigshafen, Germany;
Department of Astronomy, University of Florida, Gainesville, FL 32611; kallrath@zx.basf-ag.de

E. F. MILONE

RAO, Deptartment of Physics and Astronomy, The University of Calgary, 2500 University Drive N.W., Calgary, Alberta T2N1N4, Canada;
milone@acs.ucalgary.ca

DIRK TERRELL

Department of Astronomy, University of Florida, Gainesville, FL 32611; terrell@astro.ufl.edu

AND

ANDREW T. YOUNG

Department of Astronomy, San Diego State University, San Diego, CA 92182-1221; aty@sdsu.edu

ABSTRACT

We summarize recent improvements to a version of the Wilson-Devinney program that is widely used for the analysis of eclipsing binary data, and we describe the new WD95 program. WD95 contains the University of Calgary version of the Wilson-Devinney code, which supports the use of the Kurucz atmosphere models; it provides options to use multiple epoch data and multiwavelength synoptic passbands. The WD95 program contains an improved input/output interface, simplex algorithms for initial searches and tests, and versions of Wilson-Devinney DC and LC programs and options to switch to automatic differential corrections or a damped least-squares solver using normal equations that are modified as per the Levenberg-Marquardt scheme. This paper describes some tests of the damped least-squares solver with simulated data.

Subject headings: binaries: eclipsing — methods: numerical

1. INTRODUCTION

Now a quarter-century old, the Wilson-Devinney program is the most widely used general light-curve modeling code (McNally 1991).

Starting around 1990, several improvements were made to the original Wilson-Devinney program by a group organized by Milone and centered at the University of Calgary and by Kallrath at the University of Bonn, ultimately embodied in the software codes WD93K93 and LC93KS, respectively. The successor to these programs is WD95, described in § 5 of this paper.

When a light-curve model is used to analyze eclipsing-binary data, the goal is the solution of the "inverse problem," viz., the determination of a set of elements (in light-curve parlance, parameters) that describe and define a light curve. The inverse problem leads to a nonlinear least-squares problem that is very often solved by differential corrections. This method was first used to determine the parameters of eclipsing binary systems by Wyse (1939) and Kopal (1943). Many authors (Wilson & Biermann 1976; Kallrath 1993, among others) show that the method of differential corrections faces all the problems that usually occur in nonlinear multiparameter fitting: parameter correlation, ill-conditioned normal equations, divergence due to nonlinearities in the underlying light-curve model and the uniqueness problem; i.e., several local minima may exist in parameter space (see Kallrath & Milone 1998). Some problems (e.g., finite-difference approximations to the partial derivatives or divergence problems caused by bad initial-parameter guesses) can be overcome by gradient-free optimization methods utilizing direct search algorithms (see, for instance, Murray 1972; Lootsma 1972). An efficient representative of such procedures is the simplex algorithm (Spendley, Hext, & Himsworth 1962; Nelder & Mead 1965), which was first used for light-curve analysis by Kallrath & Linnell (1987). Since then, several authors have successfully applied this method to light-curve analysis; see Kallrath (1993) for a review of the topic and the literature.

The ideal light-curve modeling software would combine the simplex algorithm for initial parameter search with a stable local derivative based procedure for accurate and efficient determination of the solution. The development and implementation of a damped differential-correction method into the WD program can be seen as a step in this direction. The Levenberg-Marquardt algorithm—sometimes also called Marquardt algorithm, since it was proposed independently by Levenberg (1944) and Marquardt (1963)—is such a method. It has already been used in light-curve analysis by Hill (1979) in his programs LIGHT and LIGHT2, and Wilson (1997, private communication), although not using it as an automated routine, reports positive experience with this technique as well.

2. MODEL ATMOSPHERES

Our version of the WD code, WD93K93, which was developed at the University of Calgary and has been used for light-curve modeling of eclipsing-binary star systems for the past several years, makes use of Kurucz's (1993, private communication) stellar-atmosphere models contained on a CD-ROM kindly provided by R. Kurucz.

The Kurucz models have been integrated over a number of passbands: the $UBVR_JI_J$, R_CI_C, and $ubvy$ optical passbands; the Johnson (1966) $JHKLMN$ and a set of passbands that have far less sensitivity to water vapor (and that should be far more transformable between local systems),

namely, iz, iJ, iH, iK, iL, iL', iM, iN, in, and iQ, in the infrared; and, over a series of ranges of wavelength in the far ultraviolet that are appropriate for *IUE, Hubble Space Telescope,* or other rocket-ultraviolet passbands. The details of the improved infrared passbands may be found in Young et al. (1991). The integration is carried out across the full range of Kurucz atmosphere models for a given chemical composition, resulting in tables of T versus $\log g$ of atmospheric flux divided by blackbody flux.

The programs WD93K93 and LC93KS require the $\log g$ input values and lists of the passband tables of flux ratios in which the input (and adjusted) temperatures $T_{1,2}$ and $\log g_1, g_2$ are interpolated. These additional input variables are also required in WD95.

Further code improvements were discussed at a workshop held in Calgary in 1995 July, directed primarily at improving program efficiency and solution determination. The basic improvement was the use of damped least squares, described in the next section. Since we are not the first to use the Levenberg-Marquardt algorithm for light-curve modeling work (see, e.g., Hill & Rucinski 1993, who describe the usage of the algorithm as it was applied in the program LIGHT2), only a brief review of the properties of this algorithm is necessary here.

3. A DAMPED LEAST-SQUARES ALGORITHM

3.1. *Determination of Eclipsing-Binary Parameters*

The determination of physical parameters from eclipsing-binary light curves or radial velocity curves is called the "inverse problem," which can be formulated as a nonlinear least-squares problem. The aim is to minimize the deviations between the theoretical or calculated curve and the observations. The parameter vector x that produces minimum deviation is the system solution; the corresponding calculated light curve is said to be the *best fit* to the data. A measure of the deviation is the sum of the squared residuals and may be taken either weighted (indicating the true effect on the light-curve solution) or unweighted (to indicate the quality of the data). To have a measure that is independent of the number m of free parameters or the number n of data points, the standard deviation of the fit, σ_{fit}, the standard error of a single observation of unit weight derived from the fit, should be used. For the mathematical formalism of the inverse problem the reader is referred to Kallrath & Milone (1998).

3.2. *Damped Differential Corrections*

The damped least-squares (DLS) algorithm now implemented in WD95 is based on the normal equations and modification of the matrix $\mathbf{C}$ in equation (1), and it is almost identical to the Levenberg-Marquardt scheme described in Press et al. (1992).

The basic idea is to replace the normal equations (eq. [1]) of the linear least-squares problem

$$\mathbf{C}\,\Delta x_k = -\mathbf{A}(x_k)\mathbf{W}d(x_k)\,, \quad \mathbf{C} \equiv \mathbf{A}(x_k)\mathbf{W}\mathbf{A}^T(x_k)\,, \quad (1)$$

where x_k is the initial solution present at the beginning of the kth iteration, Δx_k is the correction vector defined in such a way that $x_k + \Delta x_k$ gives the vector in the next iteration, $\mathbf{A}$ is the Jacobi matrix of the least-squares function $f(x)$, $\mathbf{W}$ are weights, and $d(x_k)$ is the current residual vector, by

$$[\mathbf{C} + \mathbf{D}]\Delta x_k = -\mathbf{A}(x_k)\mathbf{W}d(x_k)\,, \quad (2)$$

where $\mathbf{D}$ is a diagonal matrix.

Obviously, this is equivalent to adding a set of observations of the form

$$\mathbf{D}\Delta x_k = 0 \quad (3)$$

to our original system, bringing the correction vector Δx_k toward zero. The damped solution Δx_k augments the function to be minimized, $f(x)$, by the squared length of Δx_k, the components of which are weighted by the elements of $\mathbf{D}$. For a fixed step length $|\Delta x_k|$, the damped solution gives the lowest possible value of $f(x)$ on a sphere with radius Δx_k and center x_k, if $\mathbf{D} = \lambda\mathbf{1}$. As the damping becomes larger, the solution vector approaches the gradient $\mathbf{V}_x[f(x)]$ at x_k. The damped least-squares method interpolates between the search direction determined as the gradient or the Gauss-Newton direction. Thus it combines the stability of gradient methods far from the solution with the convergence of Newton methods near the end, if the damping is properly changed during the iterations. The crucial part of the method is to choose the damping properly. The most successful damping schemes normalize $\mathbf{C}$ and then look for the minimum of the one-dimensional function $\phi(d) \equiv f[x_k + \Delta x_k(\lambda)]$ at each iteration.

Both linear and nonlinear problem approaches are in trouble when $f(x)$ has long, narrow valleys. Nonlinearity bends the valleys; collinearity makes them elongated. In such cases, the path of the solution vector $x_k(\lambda)$ tends to run obliquely into the valley wall instead of down its center. Thus we want to eliminate such features if we can. Correlations between parameters x cause diagonal valleys in $f(x)$ and can only be eliminated by changing variables to rotate axes in parameter space. However, valleys parallel to the axes can be made circular by rescaling the variables. This need not be done explicitly: Meiron (1965) shows that multiplying the diagonal elements c_{jj} by $(1 + \lambda)$, as originally suggested by Levenberg (1944), is equivalent to scaling the variables by $(c_{jj})^{1/2}$, adding λ to the scaled diagonal elements (which are all unity), and transforming the scaled Δx_k back to the original system.

Thus, multiplicative damping automatically eliminates elongations of the surfaces $f(x) = $ const parallel to the x axes of parameter space. Since the axial ratios of these (hyperellipsoidal) surfaces are the ratios of the square roots of the eigenvalues λ_i of $\mathbf{C}$, the scaling introduced by multiplicative damping equalizes, as far as possible, the λ_i. Thus, this method improves the conditioning of the scaled matrix $\mathbf{C}$. That can also be seen from the following argument: Ill-conditioning of a matrix is equivalent to the presence of small eigenvalues close to zero, which in the case of the normal equations would generate large parameter corrections. These are rejected when the value of the damping constant λ is increased. Sometimes the numerical computation of the derivative even leads to negative eigenvalues in the matrix $\mathbf{C}$. For sufficiently large λ, the modified matrix $\mathbf{C} + \mathbf{D}$ is positive definite, and very small eigenvalues are cut-off.

Although there are analytical considerations for the optimum selection of the damping constant, DLS algorithms usually involve heuristic processes. Many problems that require damping, and light-curve analysis can be considered one of them, converge for one or two undamped iterations before diverging. Therefore, as suggested by Press et al. (1992), we have developed the following procedure for WD95:

1. For the initial value of the damping constant λ, pick a reasonable value, say, something between $\lambda_0 = 10^{-4}$ and $\lambda_0 = 10^{-8}$ and set $\lambda = \lambda_0$.

2. For given x_k, compute $f(x)$ and establish the normal equations; in particular, compute C.

3. Modify the diagonal elements $c_{jj} \rightarrow (1 + \lambda)c_{jj}$.

4. Solve the modified normal equation (2) to get Δx_k and compute $x' = x_k + \Delta x_k$.

5. Compute $f(x')$.

6. Apply an acceptance test: if $f(x') \leq f(x_k) \Rightarrow x_{k+1} = x'$ and $\lambda \rightarrow 0.1\lambda$, then go to step 2, or

7. if $k = 1$, set $\lambda = 1$ and then go to step 2, and

8. if $k \geq 2$, set $\lambda \rightarrow 3\lambda$ and go to step 2.

The scheme proves to be robust. Performance of the algorithm does not change too much if in step 8 the factor 3 is replaced by 5 or 10. As expressed also by Wilson (1997, private communication), accepted parameter corrections do not depend too critically on λ if λ stays in a certain region of moderate damping, say, 10^{-4} to 10^{-6}. If $\lambda \geq 0.01$, the steps Δx_k become smaller than 10^{-4}.

In WD95 all of the termination criteria described in Kallrath & Milone (1998) are implemented. Any of the criteria can be activated in the control file (∗.inf) with a software on-off switch. If any of the activated criteria are met in an iteration, the run is halted and the output file (∗.log) indicates which criterion triggered the halt.

Unfortunately, in many cases the residuals are not normally distributed about a zero mean, nor are they statistically independent, with the consequence that the errors derived from the covariance matrix lose the clear meaning they would have in the case of normally distributed residuals. Moderate deviations from normality can be handled by robust-estimation techniques, but correlated residuals represent a loss of effective degrees of freedom and are more difficult to deal with. Consequently, analyses are incomplete until a plot of the light curve and the residuals provides evidence of successful fitting.

4. TESTS OF THE DAMPED LEAST-SQUARES ALGORITHM

In this section we describe two out of seven tests of the method and program with well-defined test scenarios. In Milone et al. (1999) the method is applied to AI Phoenicis.

The program was tested by applying it to various types of synthetic data to which various amounts of Gaussian noise had been added. After a parameter set x had been chosen, the LC program was used to generate 200 data points; Gaussian noise with a specified standard deviation, σ, was then added to the light values, with no noise added to the phases. These degraded data were then compared with the uncontaminated data to determine the true noise level. The value found was then used as the target noise σ^N. The tests covered several cases where both the astrophysical parameters and the noise level were varied. In terms of the astrophysical parameters, the tests included total and partial eclipses as well as systems of different morphological types (viz., detached, semidetached, and overcontact). Only raw data were used in the present series of tests; i.e., no binned or normal points were used in the main tests. The first test, not reported here in detail, was to reproduce the original parameters if only uncontaminated data were used. This test demonstrated that the damping avoids divergence, and that the algorithm is stable.
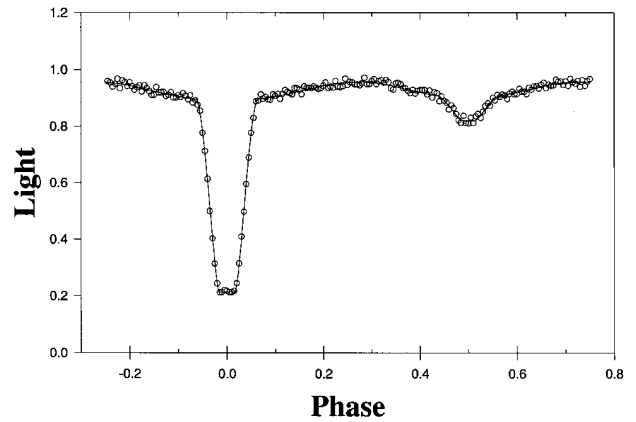

Fig. 1.—Scenario 1, light curve: data points generated for a semi-detached system with a small amount of Gaussian noise. The fitted curve shows the best WD95 fit to these data.

4.1. Scenario 1: SD Model with Low Noise Content

In this test, a semidetached binary with a total primary eclipse was modeled. The system represents a typical Algol-type binary. In this case it was assumed that the mass ratio $q = 0.21$ was available from, e.g., spectroscopy, and thus it is not a free parameter. The data had Gaussian noise $\sigma = \sigma^N \equiv 7.1 \times 10^{-3}$. As for other versions of the WD program (Wilson 1998), the component-dependent parameters are assigned subscripts 1, for the eclipsed star at the designated primary minimum, and 2, for the eclipsing star at that minimum. The adjusted parameters were inclination i, temperature T_2 of the secondary in units of 10,000 K, the Roche potential Ω_1, and the luminosity L_1 of the primary in the units used in the Wilson-Devinney program (Wilson 1998); for completeness we also give the values of $\Omega_2 = 2.2725$ and $L_2 = 3.2005$ for the solution x_* and $T_1 = 12,500$ K. The initial damping constant was set to $\lambda_0 = 10^{-8}$. The results are seen in Table 1 and the fitting in Figure 1.

In this test case all parameters are recovered within 2 σ. The deviations for the parameters i, T_2, and Ω_1 are slightly larger than in a similar scenario with $\sigma = \sigma^N \equiv 1.4 \times 10^{-2}$. But the deviation in L_1 is decreased by almost a factor 5 compared to the larger noise scenario. An inspection of the correlation matrix displayed in Table 2 shows strong correlations among i, T_2, and Ω_1: in this, and also in the second scenario, the true parameters are designated x, the starting parameters x_0, the final values x_*, the deviation $\delta = |x_* - x|$, the probable errors of the final values ε^p, and 2 σ (where σ is the standard deviation, more properly called the "mean standard error" or m.s.e.) of the parameters. The ε^p

TABLE 1

RESULTS OF THE SEMIDETACHED SCENARIO WITH LOW NOISE CONTENT

	i	T_2	Ω_1	L_1
x	89°083	0.6238	7.4858	8.5000
x_0	87°000	0.6400	8.4858	8.2331
x_*	89°682	0.6294	7.3670	8.4876
δ	0°599	0.0056	0.1178	0.0124
ε^p	0°969	0.0020	0.0488	0.0101
2 σ	1°797	0.0168	0.1464	0.0303
	$\delta < \varepsilon^P$	$\delta < 2\sigma$	$\delta < 2\sigma$	$\delta < 2\sigma$

TABLE 2

CORRELATION BETWEEN PARAMETERS

	i	T_2	Ω_1	L_1
i	1.0000000	**0.8715275**	**−0.8393344**	0.2214933
T_2	**0.8715275**	1.0000000	**−0.9665731**	0.1993583
Ω_1	**−0.8393344**	**−0.9665731**	1.0000000	−0.4164473
L_1	0.2214933	0.1993583	−0.4164473	1.0000000

values were derived from $\mathbf{C}^{-1}$ computed in the WD95 undamped differential corrections subroutine. Here, the inclination i is recovered within the probable error, while the probable errors for T_2, Ω_1, and L_1 underestimate the true errors. However, recalling that $\varepsilon^p = \frac{2}{3}\sigma$, we note that the final values of these parameters are less than 2 σ from the true values of these parameters. Confidence limits based on rigorous statistical analysis may help to overcome this problem; the use of such limits is planned for future versions of WD95.

But let us try to interpret the result. Have we stopped the iterations too early? In terms of σ_{fit}, certainly not. An inspection of the correlations between the parameters shows that T_2 and Ω_1 are strongly correlated to each other. So, each differential change in T_2 can be mimicked by a differential change in Ω_1. There is no chance of overcoming this problem. What this scenario *can* demonstrate is that the damping avoids divergence and overcomes problems associated with correlations and that the algorithm is stable.

4.2. *Scenario 2: Detached Model Including Radial Velocity Data*

This test modeled a detached system (mode = 2) with a period of 5.4592 days, a total secondary eclipse, and low-noise observations, $\sigma < 5 \times 10^{-5}$ in the photometric data in the V band. Radial velocity curves of both components were included in the analyses with $\sigma^N \equiv 5$ km s^{-1}. Each curve contained 200 data points at equidistant phases. When computing σ_{fit} for the exact parameter set, we got
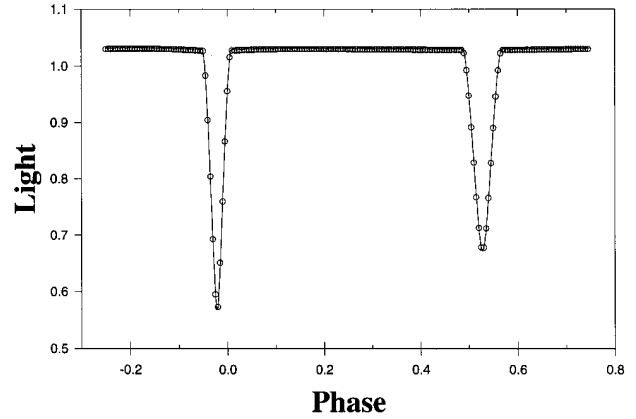

FIG. 2.—Scenario 2, light curve: data points generated for a detached system, with no noise added to the data. The fitted curve shows the best WD95 fit to these data.

$\sigma_{\text{fit}}(x) = 8.13 \times 10^{-5}$. Note that this accuracy greatly exceeds that obtainable at present telescopes. Among the fixed parameters we mention $q = 0.83$, $T_1 = 6500$ K, and $L_2 = 4.8156$. In the set of adjustable parameters we have the semimajor axis, a, in units of solar radii, the eccentricity, e, the longitude of the periastron, ω, and the system's velocity, γ; for the unit convention of γ see Wilson (1998). The success of the light curve and radial velocity fittings are seen in Figures 2 and 3, respectively.

Table 3 and 4 show the results and the correlations between parameters.

The solution was essentially reached after 10 iterations, yielding a standard deviation of 10^{-3}. The iterations 11–37 with large damping did not change the parameters by more than 10^{-3}. The standard deviation $\sigma_{\text{fit}}(x_*) = 8.13 \times 10^{-4}$ is much larger than $\sigma_{\text{fit}}(x)$, and most deviations δ are larger than 2 σ. Many other test runs produced similar results. The Levenberg-Marquardt scheme always ended up with very high values for λ; usually, $0.01 < \lambda < 1$. This behavior, even

TABLE 3

DETACHED MODEL WITH RADIAL VELOCITY DATA

	a	e	ω	γ	i	T_2	Ω_1	Ω_2	L_1
x	10.7560	0.18	65°.00	0.0583	88°.000	0.6266	9.5858	9.67990	8.0000
x_0	13.7560	0.19	64°.00	0.0883	86°.000	0.6000	7.5858	7.67990	9.0000
x_*	10.7898	0.1734	63°.90	0.0584	87°.943	0.6265	9.57771	9.57000	7.9289
δ	0.0338	0.0066	1°.10	0.0001	0°.057	0.0001	0.02862	0.04262	0.0711
ε^p	0.0065	0.00011	0°.00034	0.0002	0°.006	0.00003	0.00556	0.00750	0.0093
2σ	0.0440	0.00033	0°.00102	0.0006	0°.018	0.00009	0.01668	0.02250	0.0279
	$\delta < 2\sigma$	$\delta > 2\sigma$	$\delta > 2\sigma$	$\delta < \varepsilon^p$	$\delta > 2\sigma$	$\delta \approx 2\sigma$	$\delta > 2\sigma$	$\delta > 2\sigma$	$\delta > 2\sigma$

TABLE 4

CORRELATION BETWEEN PARAMETERS

|	a	e	ω	γ	i	T_2	Ω_1	Ω_2	L_1
a	1.000	−0.008	−0.010	0.004	−0.011	0.002	0.009	−0.012	−0.011
e	−0.008	1.000	**0.968**	−0.004	−0.015	−0.090	0.154	0.081	−0.014
ω	−0.010	**0.968**	1.000	−0.002	0.189	−0.040	−0.017	0.287	0.180
γ	0.004	−0.004	−0.002	1.000	0.009	−0.000	−0.009	0.008	0.009
i	−0.011	−0.015	0.189	0.009	1.000	−0.093	**−0.887**	**0.980**	**0.979**
T_2	0.002	−0.090	−0.040	−0.000	−0.093	1.000	0.092	−0.076	−0.149
Ω_1	0.009	0.154	−0.017	−0.009	**−0.887**	0.092	1.000	**−0.840**	**−0.944**
Ω_2	−0.011	0.081	0.287	0.008	**0.980**	−0.076	**−0.840**	1.000	**0.968**
L_1	−0.011	−0.014	0.180	0.009	**0.979**	−0.149	**−0.944**	**0.968**	1.000
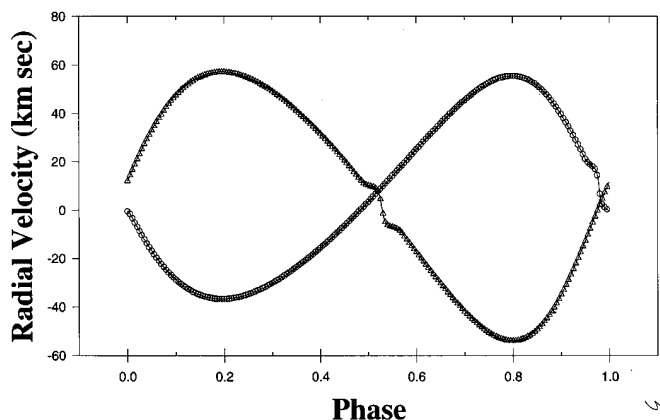
FIG. 3.—Scenario 2, radial velocity curves: data points for this double-lined spectroscopic binary generated for a detached system, but, again, without the addition of noise. The fitted curve shows the best WD95 fit to these data.

with the very accurate given data, can be explained as resulting from two reasons. First, at the exact parameter set x, there are strong correlations among certain parameters, such as between e and ω, and, even more strongly, among i and T_2, Ω_1, Ω_2, and L_1. The correlations require large damping because otherwise the parameter corrections Δx would be too large. A second reason is the limiting finite accuracy of the derivatives.

In addition to these scenarios, we have run five more, covering semidetached and overcontact models with various levels of Gaussian noise and the real binary case of AI Phoenicis. In all synthetic cases the initial parameters were satisfactorily recovered, and the AI Phoenicis case tested gave recovered results previously obtained with our WD code, WD93K93.

4.3. Results and Interpretations of the Tests

The tests show that the iterated damped least-squares method is capable of producing converged solutions, given well-conditioned data with varying amounts of Gaussian noise σ. Each solution produced a standard deviation, σ_{fit}, which was smaller than a specified amount of noise, σ^N, in the data. This is all that one could expect, and it produced quite satisfactory fits. The differences between the true and derived values of the parameters were typically within about 2 σ (in terms of the uncertainties in the parameters). In some cases that were started with initial parameter guesses very far from the solution, the damped least-squares approach failed. In such cases we did an initial search with the simplex algorithm and then continued from the best vertex with damped least squares.

Thus far, no comment has been made about the CPU time needed for the iterations. All calculations were performed on a Pentium laptop and took many hours, typical of multiple-iteration procedures with many parameters. To make comparisons with noniterative versions of the WD differential corrections program, one would need to consider the time taken to (*a*) review the parameter adjustments and their probable errors, (*b*) decide on the subset to be adjusted, (*c*) apply the adjustments to the selected parameters in a new input file, and (*d*) resubmit. While some of these procedures can be and have been automated, the selection process needs to be reviewed at least occasionally;

the timing of such a reviewing process must vary from run to run. Consequently, it is impossible to generalize, especially since one usually has other things to do between run submissions, making dead time the most likely sink of time for single iteration operations.

The important point to be made, however, is not that the present procedure is intrinsically speedier; computational time per se is not the critical issue. What counts most is having a stable procedure that can iterate automatically. This we now claim to have provided. Nevertheless, a word of warning is appropriate at this point. The algorithm is not an infallible black box with which one can produce automatic and necessarily correct solutions. Setting the stopping criteria requires experience on the part of the user so that the convergence process is not terminated too early. It is difficult to fully automate this procedure to meet all contingencies.

We have also conducted preliminary tests that included spot parameters. For the purpose of automatic iterations, the accuracy of spot modeling is not high enough in the current Wilson-Devinney program (Wilson 1995, private communication); however, further testing is planned in the near future.

5. FEATURES OF THE WD95 PROGRAM

The current capabilities of WD95 are summarized in the following:

Full functionality of the WD program, extended to double precision.
Kurucz atmospheres for a large group of passbands.
Automatic changing of limb-darkening as temperature changes.
Recalculation of log g for interpolation in the atmospheres tables when radial velocity data are present.
Parameter space searches with the simplex algorithm.
Manual and automatic differential correction options.
Levenberg-Marquardt algorithm.
Shell-based input/output interface.
A regularly updated log file with progress reports on convergence.

The program can emulate the original subroutines LC and DC. In particular, it is possible to use the method of differential corrections as in the original Wilson-Devinney program. Although the damped least-squares solver in WD95 is less dependent on the quality of guessed initial parameters than the stand-alone version of the WD program, WD93K93, we recommend that the initial parameters be chosen carefully and the simplex option be used in initial searches. Once a parameter set is close to that of the solution, a switch to the Levenberg-Marquardt algorithm accelerates the rate of convergence. A final, undamped run will provide the probable errors of the parameters. We recommend this procedure generally as an efficient way to find the likely minimum in parameter space and to achieve a successfully converged light-curve solution.

6. CONCLUSIONS

We have implemented and tested the Levenberg-Marquardt algorithm in the Wilson-Devinney program and applied it to several astrophysically relevant scenarios. The method is much less dependent on the accuracy of the initial parameter guesses than is the undamped differential correc-

tion method. Nevertheless, it helps if the initial parameters are as accurate as possible or taken from initial searches with the simplex algorithm. In our test scenarios with varying data quality, we started from bad initial guesses and recovered the original parameters *almost* always. So even in such cases we have a stable procedure that can iterate automatically and put the solutions of light-curve analyses on a more objective basis.

REFERENCES

Hill, G. 1979, Publ. Dom. Astrophys. Obs. Victoria, 15, 297
Hill, G., & Rucinski, S. M. 1993, in Modeling Eclipsing Binaries, ed. E. F. Milone (New York: Springer), 135
Johnson, H. L. 1966, ARA&A, 4, 193
Kallrath, J. 1993, in Modeling Eclipsing Binaries, ed. E. F. Milone (New York: Springer), 39
Kallrath, J., & Linnell, A. P. 1987, ApJ, 313, 346
Kallrath, J., & Milone, E. F. 1998, Eclipsing Binary Stars: Modeling and Analysis (New York: Springer), chap. 4
Kopal, Z. 1943, Proc. Am. Philos. Soc., 36, 342
Levenberg, K. 1944, Q. Appl. Math., 2, 164
Lootsma, F. 1972, Numerical Methods for Nonlinear Optimization (London: Academic Press)
Marquardt, D. 1963, SIAM J. Appl. Math., 11, 431
McNally, D., ed. 1991, Rep. Astron. Symp. Theory Comput. 21A, in Close Binary Stars, The Netherlands, IAU

Meiron, J. 1965, J. Opt. Soc. Am., 55, 1105
Milone, E. F., Kallrath, J., Terrell, D., & Young, A. T. 1999, in preparation
Murray, W. 1972, Numerical Methods for Unconstrained Optimisation (London: Academic Press)
Nelder, J. A., & Mead, R. 1965, Comput. J., 7, 308
Press, W. H., Flannery, B. P., Teukolsky, S. A., & Vetterling, W. T. 1992, Numerical Recipes (2d ed.; Cambridge: Cambridge Univ. Press)
Spendley, W., Hext, G., & Himsworth, F. 1962, Technometrics, 4, 441
Wilson, R. E. 1998, in Reference Manual to the Wilson-Devinney Program, Computing Binary Star Observables, version 1998 (Gainesville, FL: Univ. Florida)
Wilson, R. E., & Biermann, P. 1976, A&A, 48, 349
Wyse, A. 1939, Lick. Obs. Bull., 496, 17
Young, A. T., et al. 1991, PASP, 103, 221