COMPUTER SCIENCE
TECHNICAL REPORT

Time-stepping methods that favor positivity for atmospheric chemistry modeling

Adrian Sandu

CSTR-0001
September 2000

Michigan Technological University, Houghton, MI 49931

Time-stepping methods that favor positivity for atmospheric chemistry modeling

Adrian Sandu*

July 21, 2000

Abstract

Chemical kinetics conserves mass and renders non-negative solutions; a good numerical simulation would ideally produce mass balanced, positive concentration vectors. Many time stepping methods are mass conservative; however, unconditional positivity restricts the order of a traditional method to one. The projection method presented in [13] post-processes the solution of a one-step integration method to ensure mass conservation and positivity. It works best when the underlying time-stepping scheme favors positivity. In this paper several Rosenbrock-type methods that favor positivity are presented; they are designed such that their transfer functions together with several derivatives are nonnegative for all real, negative arguments.

Keywords: Chemical kinetics, linear invariants, positivity, numerical time integration.

1 Introduction

Air quality models [3, 11] solve the convection-diffusion-reaction set of partial differential equations which describe the atmospheric physical and chemical processes. Usually an operator-split approach is taken: chemical equations and convection-diffusion equations are solved in alternative steps. In this setting the integration of chemical kinetic equations is a demanding computational task. The chemical integration algorithm should be stable in the presence of stiffness; ensure a modest level of accuracy, typically 1%; preserve mass; and keep the concentrations positive.

Most popular ODE integrators (multistep, Runge-Kutta, Rosenbrock) preserve mass, but positivity is more difficult to achieve. Unconditional positivity restricts the order of a numerical method to one [2]. Clipping (setting the negative concentrations to zero) enhances stability but artificially adds mass to the system. Solution projection and stabilization [13] are postprocessing techniques which allow a positive, mass balanced solution without restrictions on the order or time step of the integration method.

In order to be effective, projection and stabilization must be paired with time stepping methods that favor positivity. Although positivity is not guaranteed, these methods tend to keep nonnegative concentrations and reduce the postprocessing overhead. One example of positivity favorable method is Ros2 (advocated by Verwer et. al. [15]).

In this paper we explore several methods that favor positivity. These methods are developed based on the conjecture that good positivity behavior is related to the positivity of the transfer function and its first several derivatives for negative real arguments. All methods are of Rosenbrock type. Numerical experiments show that the new methods paired with solution projection produce smaller overheads.

*Department of Computer Science, 205 Fisher Hall, Michigan Technological University, 1400 Townsend Drive, Houghton, MI 49931, Phone: (906)-487-2187, Fax: (906)-487-2283, E-mail: asandu@mtu.edu

The paper is organized as follows. Section 2 reviews the properties of chemical kinetic systems. Positive time stepping methods are discussed in Section 3 (following Bolley and Crouzeix [2]), and in Section 4 (solution postprocessing techniques). Section 5 presents new time stepping methods that favor positivity. A test case from stratospheric chemistry is considered in Section 6, where different numerical results are presented. Finally, the findings and conclusions of the paper are summarized in Section 7. Appendix A defines the numerical integration methods and Appendix B the test chemical mechanism.

2 Mass action kinetics, linear invariants and positivity

Consider a chemical kinetic system with N species $y_1, \cdots, y_N$ interacting in r chemical reactions. The time evolution of the chemical concentrations is governed by the "mass action kinetics" differential law[1]

$$y' = S \cdot \omega(y) \ , \quad y(t_0) = y^0 \ , \tag{1}$$

where $S \in \Re^{N \times r}$ is the matrix of stoichiometric coefficients and $\omega \in \Re^r$ is the vector of reaction speeds. Any solution of (1) satisfies

$$A^T\, y(t) = A^T\, y^0 = b = \text{const.} \qquad \text{for all } t \geq t_0 \ , \tag{2}$$

where $A \in \Re^{N \times m}$ is a matrix whose columns span $\ker\left(S^T\right)$, i.e. $A^T S = 0$. This implies $A^T y'(t) = 0$ and $A^T y(t)$ is invariant in time. The vector $b \in \Re^m$ contains the m invariant values. Simply stated, the existence of linear invariants ensures that mass is conserved during chemical reactions.

Let us now separate the production terms $P(y)$ from the destruction terms $D(y)$ in (1)

$$y_i' = P_i(y) - D_i(y)\, y_i \ , \qquad 1 \leq i \leq N \ .$$

The special form of the reaction speeds ensures that $P_i(y)$ and $D_i(y)$ are polynomials in y with positive coefficients. If at time moment τ all concentrations are nonnegative, $y(\tau) \geq 0$, and the concentration of species i is zero, $y_i(\tau) = 0$, then the corresponding derivative is nonnegative, $y_i'(\tau) = P_i(\tau) \geq 0$, which implies that

$$y(t_0) \geq 0 \quad \Longrightarrow \quad y(t) \geq 0 \quad \text{for all } t \geq t_0 \ . \tag{3}$$

In short, the concentrations cannot become negative during chemical reactions.

Linear invariants (2) and positivity (3) imply that the solution of (1) remains all the time within the reaction simplex

$$y(t) \in \mathcal{S} \text{ for all } t \geq t_0 \ , \quad \mathcal{S} = \left\{ y \in \Re^s \text{ s.t. } A^T\, y = b \text{ and } y \geq 0 \right\} \ . \tag{4}$$

A general principle in scientific computing says that the numerical solution must capture (as much as possible) the qualitative behavior of the true solution. Good numerical methods for integrating chemical reaction models (1) should therefore *be unconditionally stable*, as the system is usually stiff (this requires implicit integration formulas); should *preserve the linear invariants*, otherwise artificial mass sources (or sinks) are introduced; and should *preserve solution positivity.*

Negative concentrations are non-physical. In addition, the kinetic system may become unstable for negative concentrations, as shown by Verwer et. al. [15]. An operator-split solution of convection-diffusion-reaction atmospheric equations alternates chemical integration steps with advection steps; negative concentrations from chemical integration can hurt the positivity of the following advection step, which will perturb the next chemical step etc. leading to poor quality results.

It is well known that the most popular integration methods (Runge-Kutta, Rosenbrock and Linear Multistep) preserve exactly[2] all the linear invariants of the system [15]. Moreover,

[1] We denote by y_i both the chemical species i and its mass concentration.

[2] If the computations are performed in infinite arithmetic precision.

the (modified) Newton iterations used to solve for implicit solutions also preserve the linear invariants at each iteration. With linear-preserving integration methods the accuracy of the individual components is given by the truncation errors (e.g. having a magnitude 10^{-4}), while the accuracy of the linear invariants is only affected by the roundoff errors (having a much smaller magnitude, e.g. 10^{-14}).

3 Positive time-stepping methods

Positivity of the numerical solution is more difficult to achieve. We would like to obtain higher order methods that unconditionally preserve positivity. The results of Bolley and Crouzeix [2] show that unconditional positivity limits the order of the numerical method to one. For convenience we briefly review their analysis. Bolley and Crouzeix [2] focus on linear systems

$$y' \;=\; A\,y \quad \text{with} \quad A \in \mathcal{M}_\alpha \;, \quad y(t_0) = v \geq 0 \;, \tag{5}$$

where

$$\mathcal{M}_\alpha = \{ A \in \Re^{n\times n} \;:\; A_{ij} \geq 0 \text{ for } i \neq j \text{ and } A_{ii} \geq -\alpha \;, \forall i \}$$

A one step method of order p, applied to (5) gives

$$y^{n+1} = R(hA)\, y^n \;,$$

where $R(z)$ is a rational function whose Taylor expansion around zero matches the expansion of e^z up to z^p. A necessary and sufficient condition for the scheme to be positive is that $R(hA) \geq 0$.

Let $M = -h\alpha I$ and $N = h(\alpha I + A)$, such that $M + N = hA$ and $N \geq 0$. Bolley and Crouzeix established that the following Taylor type expansion holds

$$R(M+N) = \sum_{j\geq 0} \frac{1}{j!} R^{(j)}\,(M) \cdot N^j = \sum_{j \geq 0} \frac{h^j}{j!} R^{(j)}\,(-h\alpha) \cdot (\alpha I + A)^j \;. \tag{6}$$

Let γ_R be the largest positive number such that $R(z)$ and all its derivatives $R'(z)$, $R''(z)$, $\cdots$ are nonnegative for $z \in [-\gamma_R, 0]$. We say that R is *absolutely monotonic* on $[-\gamma_R, 0]$. Since $\alpha I + A \geq 0$, it is clear that

$$R(hA) \geq 0 \;\text{ for all }\; A \in \mathcal{M}_\alpha \quad \Leftrightarrow \quad h \leq \frac{\gamma_R}{\alpha} \;. \tag{7}$$

For (linearized) chemical kinetics $1/\alpha$ is roughly the lifetime of the fastest species in the system. Consequently, the positivity step size restriction (7) is similar to the upper bound required for the stability of an explicit integration scheme. This leads to impractically small steps for stiff chemical systems.

In practice we need a numerical method which unconditionally preserves positivity: $A \in M_\alpha$ $\Rightarrow R(hA) \geq 0$ for all h. From (7) this method must have a stability function with $\gamma_R = \infty$, i.e. $R(z)$ must be absolutely monotonic on $(-\infty, 0]$. But this condition prohibits R of approximating e^z to more than first order (see [2]). Therefore there is no one-step method which unconditionally conserves positivity of order greater than or equal to two.

Hundsdorfer [8] proved that the (first order) implicit Euler method is unconditionally positive. In practice, even implicit Euler may produce negative values since the iterative solution process is halted after a finite number of steps; while the exact solution is non-negative, the successive approximations computed by (modified) Newton are not.

4 Positivity by solution postprocessing

Postprocessing corrects the computed solution at each step such that negative concentrations are avoided. The most popular method is *clipping*, which simply sets the negative components to zero. Clipping destroys the preservation of linear invariants. Moreover, all clipping errors act in the same direction, namely increase mass (artificially); therefore they accumulate over time and may lead to significant global errors over longer simulation intervals.

In [13] two methods for preserving both mass balance and positivity were developed. *Solution projection* method is based on a linear invariant-preserving one-step integration method Φ (e.g. Runge-Kutta or Rosenbrock)

$$z^{n+1} = \Phi_h^f\left(y^n\right) \ .$$

Here t^n denotes the discrete time value at $n^{\rm th}$ step, y^n the $n^{\rm th}$ step solution, $h = t^{n+1} - t^n$ the step size, and $f(t,y) = S\omega(t,y)$.

If some of the computed concentrations are negative

$$z_{i1}^{n+1} < 0 \quad \cdots \quad z_{ip}^{n+1} < 0 \ ,$$

the next step approximation y^{n+1} is the solution of the following quadratic optimization problem

$$\min \ \frac{1}{2}\left\|y^{n+1} - z^{n+1}\right\|_G^2 \quad \text{subject to} \quad A^T y^{n+1} = b \ , \ y^{n+1} \geq \epsilon \ . \tag{8}$$

If the are no negative components then $y^{n+1} = z^{n+1}$. The solution y^{n+1} is the projection of z^{n+1} onto the reaction simplex $\mathcal{S}$; ϵ are small numbers which ensure a positive solution in the presence of roundoff. The norm $\|w\|_G = \sqrt{w^T G w}$ is the typical error norm used by the ODE step size controller, with the positive matrix G given at each step by

$$G = \operatorname*{diag}_{1\leq i \leq s} \left[\frac{1}{s\left(atol + rtol\,|z_i^{n+1}|\right)^2}\right] . \tag{9}$$

In [13] it was shown that the projected vector is a better (G-norm) approximation to the true solution then is the computed vector,

$$\left\|y^{n+1} - y(t^{n+1})\right\|_G \leq \left\|z^{n+1} - y(t^{n+1})\right\|_G \ .$$

Any algorithm for quadratic programming can be employed to solve (8). We found the primal-dual algorithm of Goldfarb and Idnani [5] to be a suitable solution method.

Solution stabilization is a simpler alternative to projection. The next step solution is computed as

$$\begin{aligned} z^{n+1} &= \Phi_h^f\left(y^n\right) \ \text{ with } \ z_{i1}^{n+1}, \cdots, z_{ip}^{n+1} < 0 \ , \qquad B = \left[A \,|\, e_{i1} \,|\cdots|\, e_{ip}\right] \ , \\ y^{n+1} &= z^{n+1} - G^{-1} B \left(B^T G^{-1} B\right)^{-1} \begin{bmatrix} A^T z^{n+1} - b \\ z_{i1}^{n+1} - \epsilon_{i1} \\ \vdots \\ z_{ip}^{n+1} - \epsilon_{ip} \end{bmatrix} . \end{aligned} \tag{10}$$

It can be directly verified that the solution satisfies $A^T\, y^{n+1} = b$, $y_{i1}^{n+1} = \epsilon_{i1}, \cdots, y_{ip}^{n+1} = \epsilon_{ip}$. The method does not guarantee positivity, since the projection step may render other components negative (i.e. $y_j^{n+1} < 0$ for $j \neq i1...ip$), but can have a beneficial effect on maintaining positivity.

In [13] it was shown that projection and stabilization work best when the underlying numerical method favors positivity. Such methods are discussed next.

5 Methods that favor positivity

In the view of the theory developed in [2] we do not require the methods to be unconditionally positive, but we relax the requirements to methods which favor positivity. Although positivity is not guaranteed, these methods tend to keep nonnegative concentrations and reduce the postprocessing overhead.

In [15, 16, 17] it was noted that the second order Rosenbrock method Ros2 (16) has favorable positivity properties, and the method is stable for nonlinear problems even with large fixed step sizes. It was also noted that Ros2 provides positive solutions for the scalar problems $C' = -kC$ and $C' = -k\,C^2$, $C(t_0) \geq 0$.

In [13] we conjectured that a possible explanation for the good observed behavior is that not only the transfer function of this method, but also its first two derivatives are nonnegative for real, negative arguments (i.e. $R(z), R'(z), R''(z) \geq 0$ for any $z \leq 0$). Therefore the first (most significant) terms in the Taylor series (6) are nonnegative. Higher order terms may be negative, but they are weighted by higher powers of the step size h, therefore the negative part will hopefully be small.

Based on this conjecture we develop several methods whose transfer functions and their first several derivatives are nonnegative for any real, negative argument. Specifically, we consider one-step, second order, s-stage methods ($s \geq 2$) with a stability function of the form

$$R(z) = \frac{1 - az}{(1-\gamma z)^s} \quad \text{with} \quad a = \frac{1+\sqrt{s}}{s-1}\,, \quad \gamma = \frac{s+\sqrt{s}}{s(s-1)}\,.$$

The relations for a and γ follow from imposing R to approximate the exponential to second order. We have

$$R'(z) = \frac{1 - \gamma\,(\sqrt{s}+1)\,z}{(1-\gamma z)^{s+1}}\,, \qquad R''(z) = \frac{\gamma s\,(2 + (s-1)\gamma) - \gamma^2 s\,(\sqrt{s}+1)\,z}{(1-\gamma z)^{s+2}}\,,$$

and we see that $R(z), R'(z), R''(z) \geq 0$ for any $z \leq 0$. Ros2 falls into this category of methods for $s = 2$. For more stages these methods have the added benefit of higher damping of transients, and faster steady-state introduced for short-lived species.

We developed four such methods which, to our knowledge, have not been proposed elsewhere. The methods are second order accurate and require only two function evaluations:

- Method A (17) is three-stage, second-order, stiffly accurate. It preserves its order for inexact Jacobians;
- Method B (18) is three-stage, second-order, stiffly accurate. One of the order 3 conditions is also satisfied;
- Method C (19) is three-stage, second-order (but not stiffly accurate). It works with inexact Jacobians; One of the order 3 conditions is also satisfied.
- Method D (20) is four-stage, second-order, stiffly accurate. It allows inexact Jacobians and has an embedded method of order 3.

The coefficients for all methods are given in Appendix A.

6 Numerical Results

We consider the basic stratospheric reaction mechanism presented in Appendix B (and adapted from NASA HSRP/AESA [9]). The numerical examples are implemented in MATLAB. The simulation starts at noon with the initial concentrations of Table 1 and continues for 72 hours. The computation of G-norms was done with $rtol = 10^{-5}$ and $atol = 10^{-3}$. Throughout the tests the minimal values were set to $\epsilon_i = 1$ molec/cm^3. Reference solutions were obtained with the MATLAB integration routine ODE15S (variable order numerical differentiation formula); the control parameters were $RelTol = 10^{-8}$, $AbsTol = 10^{-8}$, with analytic Jacobian.

The integration algorithms used are BDF2 (12), Ros2 (16), Rodas3 (15), RK2 and RK2+ (13), as well as the new methods A (17), B (18), C (19), and D (20). For the Rosenbrock methods we use the non-autonomous forms. RK2 and RK2+ differ by the value of the coefficient γ; the classical formula RK2 is more accurate, while RK2+ has the same transfer function as Ros2 (hence falls in the previously discussed category of positive favorable methods).

To compare the performance of different methods we measured the solution accuracy at the end of the integration interval ($t = T_F$). With y^R the reference solution and y the computed solution, the error measure reads

$$E = \sqrt{\frac{1}{s}\sum_{i=1}^{s}\left(\frac{y_i(T_F) - y_i^R(T_F)}{y_i^R(T_F)}\right)^2}\,. \tag{11}$$

Figure 1 shows the solution accuracies (11) versus computational work (Kflops) for different methods (plain versions). All methods perform similarly, with methods A and B slightly better at low accuracies and C more performant at high accuracies. RK2+ and BDF2 are not competitive at low accuracies, presumably due to convergence problems at large step sizes. The conclusions of this diagram are limited, however, since the system is small and sparsity is not accounted for.

Figures 2, 3, and 4 show the average negative values and the percent of time negative values were obtained for O^{1D}, O and NO respectively. For O^{1D} (Figure 2) methods A, B, D give reasonably small negative concentrations. Even smaller concentrations (in absolute value) are produced by BDF2 and Rodas3. The percent of steps which produced negative O^{1D} concentrations changes for different step sizes for each of the methods, so there is no "clear winner" here. For O (Figure 3) method D shows negligible percent of negative values for steps up to 30 min. Methods A and B show small percent for small and large steps, while C and Ros2 show large percents of negative values. The mean negative values are smallest for methods A, B and D. Also RK2+ shows better results than RK2 for small step sizes. For NO (Figure 4) the methods A, B (up to 15 min), C (up to 10 min), and D (up to 30 min) show a negligible percent of negative steps. The mean negative concentrations are smallest for A, B and D. RK2+ again produces fewer times negative concentrations than Ros2, but the average negative concentrations are large.

Interestingly, the Ros2 performance is rather modest with regard to positivity. We assume that this is due to the fact that, once a concentration c^n accidentally becomes negative, the next step approximation (of the form $c^{n+1} \approx R(hJ)c^n$) will tend to remain negative precisely because of the positivity of R.

Figure 5 shows the computational overhead (percentage) of the projected versions versus the plain versions. At small time steps (large computational work and high accuracies) Ros2 has negligible overheads; methods A, B and D have overheads around 5%, while the overheads of method C are different at different step sizes. RK2+ has half the overhead of RK2. BDF2, RK2 and Rodas3 have higher overheads. At large steps (small computational work and low accuracies) Ros2 is the clear winner, with small overheads.

These experiments show a clear difference between the plain versions and the projected versions. While the plain Ros2 performance is modest with regard to positivity, the corrected version performance is excellent. This can be explained by the fact that once the negative concentration c^n are corrected, then the next step value c^{n+1} will tend to remain positive.

7 Conclusions

Projection and stabilization ensure mass conservation and positivity for the numerical solutions of chemical kinetic problems. The techniques are based on postprocessing the next-step approximations given by linear-preserving methods, should negative concentrations develop.

Both techniques have to be paired with a positivity-favorable numerical integration methods, in order to reduce the overheads. Although such methods do not guarantee positivity, they seldom produce non-positive results, which minimizes the overheads incurred by projection or stabilization. Such a method is for example Ros2 [16, 17]; Verwer et. al. explained the good properties of Ros2 by the fact that its stability function $R(z) \geq 0$ for all $z \leq 0$, and also by the fact that the result of integrating $c' = -c^2$, $c(t_0) \geq 0$ is unconditionally positive.

Based on the theory of Bolley and Crouzeix [2] we conjectured that the positivity of the first and second derivatives of the transfer function $R'(z), R''(z) \geq 0$ for all $z \leq 0$ play a role in making a method positivity favorable. Following this conjecture we derived four new Rosenbrock methods which fulfill this condition. In addition, we tested a a Runge-Kutta scheme (RK2+) whose transfer function has similar properties.

The results seem to confirm the conjecture. Solution projection gives low overheads when paired one of these positivity favorable techniques. Also, the modified RK2+ method has consistently given lower overheads than the original method RK2, even if the latter method is more accurate.

The conclusion, however, is not as clear cut as one hoped. First, method C does not behave as well as the other methods, suggesting that extra factors besides the transfer function determine positivity. Second, there is a significant difference between the behavior of the plain and the corrected methods. For example the Ros2 results are rather dissapointing (in terms of positivity) for the noncorrected version, but are excellent for the corrected version.

Appendix A. Numerical integration methods

The second order backward differentiation formula BDF2 [6, Section III.1] is

$$y^{n+1} = Y^n + \frac{2}{3}\, h\, f\left(t^{n+1}, y^{n+1}\right)\ , \quad Y^n = \frac{4}{3} y^n - \frac{1}{3} y^{n-1}\ . \tag{12}$$

Here $t^{n+1} = t^n + h = t^{n-1} + 2h$; for variable time steps the coefficients change. The very first step requires both y^1 and y^0; the former is given, while the latter is obtained with one backward Euler step.

The second order Runge-Kutta method RK2 [10] is

$$\begin{array}{rcl} y^{n+1} & = & y^n + (1-\gamma)\, k_1 + \gamma\, k2 \\ k_1 & = & h\, f\left(t^n + \gamma h, y^n + \gamma k_1\right), \\ k_2 & = & h\, f\left(t^n + h, y^n + (1-\gamma) k_1 + \gamma k_2\right), \end{array} \tag{13}$$

with $\gamma = 1 - \sqrt{2}/2$. For RK2+ the value is $\gamma = 1 + \sqrt{2}/2$.

An s-stage Rosenbrock method reads

$$\begin{array}{rcl} y^{n+1} & = & y^n + \sum_{i=1}^{s} m_i k_i\ , \qquad \hat{y}^{n+1} = y^n + \sum_{i=1}^{s} \hat{m}_i k_i\ , \\ \left(\frac{1}{\gamma h} I - J\right)\, k_i & = & f\left(t^n + \alpha_i h, y^n + \sum_{j=1}^{i-1} a_{ij} k_j\right) + \sum_{j=1}^{i-1} \left(\frac{c_{ij}}{h}\right) k_i + g_i h f_t\ , \quad i = 1, \cdots, s\ . \end{array} \tag{14}$$

where the Jacobian matrix $J = \partial f(t,y)/\partial y$ and the time partial derivative $f_t = \partial f(t,y)/\partial t$ are evaluated at $t = t^n$. A specific method is defined by its coefficients.

The Rodas3 method [12] is third order accurate and reads

$$\begin{array}{c} \gamma = 1/2\ , \quad A = \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 2 & 0 & -1 & 0 \end{array}\right]\ , \quad C = \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 1 & -1 & -8/3 & 0 \end{array}\right]\ , \\ \alpha = \left[\begin{array}{c} 0 \\ 0 \\ 1 \\ 1 \end{array}\right]\ , \quad g = \left[\begin{array}{c} 1/2 \\ 3/2 \\ 0 \\ 0 \end{array}\right]\ , \quad m = \left[\begin{array}{c} 2 \\ 0 \\ 1 \\ 1 \end{array}\right]\ , \quad \hat{m} = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \end{array}\right]\ . \end{array} \tag{15}$$

The second order Rosenbrock scheme Ros2 [16, 17] is defined as

$$\begin{array}{c} \gamma = 1 + \sqrt{2}/2\ , \quad A = \left[\begin{array}{cc} 0 & 0 \\ 2 - \sqrt{2} & 0 \end{array}\right]\ , \quad C = \left[\begin{array}{cc} 0 & 0 \\ -4 + 2\sqrt{2} & 0 \end{array}\right]\ , \\ \alpha = \left[\begin{array}{c} 0 \\ 1 \end{array}\right]\ , \quad g = \left[\begin{array}{c} 1 + \sqrt{2}/2 \\ -1 - \sqrt{2}/2 \end{array}\right]\ , \quad m = \left[\begin{array}{c} (6 - 3\sqrt{2})/2 \\ 1 - \sqrt{2}/2 \end{array}\right]\ , \quad \hat{m} = \left[\begin{array}{c} 2 - \sqrt{2} \\ 0 \end{array}\right]\ . \end{array} \tag{16}$$

The vector $y^n + (1/\gamma)k_1$ is a consistent approximation at t^{n+1} and was used to implement the error estimator in the variable step formulation.

Method A is second-order, stiffly accurate, and is defined by

$$\begin{gathered} A = \begin{bmatrix} 0 & 0 & 0 \\ 3-\sqrt{3} & 0 & 0 \\ 3-\sqrt{3} & 0 & 0 \end{bmatrix} , \quad C = \begin{bmatrix} 0 & 0 & 0 \\ -6+4\sqrt{3} & 0 & 0 \\ 3-2\sqrt{3} & 3-2\sqrt{3} & 0 \end{bmatrix} , \\ \gamma = (3+\sqrt{3})/6 \ , \quad \alpha = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} , \quad g = \begin{bmatrix} (3+\sqrt{3})/6 \\ (1+\sqrt{3})/2 \\ 0 \end{bmatrix} , \quad m = \begin{bmatrix} 3-\sqrt{3} \\ 0 \\ 1 \end{bmatrix} . \end{gathered} \tag{17}$$

It requires only 2 function evaluations and preserves its order for inexact Jacobians.

Method B is second-order, stiffly accurate, and is defined by

$$\begin{gathered} \gamma = (3+\sqrt{3})/6 \ , \quad A = \begin{bmatrix} 0 & 0 & 0 \\ 3-\sqrt{3} & 0 & 0 \\ 3-\sqrt{3} & 0 & 0 \end{bmatrix} , \quad C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -4+2\sqrt{3} & 1-\sqrt{3} & 0 \end{bmatrix} , \\ \alpha = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} , \quad g = \begin{bmatrix} (3+\sqrt{3})/6 \\ (3+\sqrt{3})/6 \\ 0 \end{bmatrix} , \quad m = \begin{bmatrix} 3-\sqrt{3} \\ 0 \\ 1 \end{bmatrix} . \end{gathered} \tag{18}$$

One of the order 3 conditions is also satisfied.

Method C is second-order (but not stiffly accurate) and is defined by

$$\begin{gathered} A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ (1272-823\sqrt{3})/354 & 3(-51+49\sqrt{3})/59 & 0 \end{bmatrix} , \\ \gamma = (3+\sqrt{3})/6 \ , \quad C = \begin{bmatrix} 0 & 0 & 0 \\ -1+7\sqrt{3}/18 & 0 & 0 \\ (25-13\sqrt{3})/6 & 12-6\sqrt{3} & 0 \end{bmatrix} , \\ \alpha = \begin{bmatrix} 0 \\ 0 \\ 2/3 \end{bmatrix} , \quad g = \begin{bmatrix} (3+\sqrt{3})/6 \\ (39+14\sqrt{3})/108 \\ (9+\sqrt{3})/6 \end{bmatrix} , \quad m = \begin{bmatrix} (-12089+5037\sqrt{3})/472 \\ 9(344-135\sqrt{3})/118 \\ 3(3-\sqrt{3})/4 \end{bmatrix} . \end{gathered} \tag{19}$$

It also requires only 2 function evaluations and is more accurate than method A, since one of the order 3 conditions is also satisfied. It works with inexact Jacobians.

The stability function for both methods A and B is

$$R(z) = \frac{1 - \frac{1+\sqrt{3}}{2} z}{\left(1 - \frac{3+\sqrt{3}}{6} z\right)^3}$$

and we have $R(z), R'(z), R''(z) \geq 0$ for $z \leq 0$.

Method D is second-order, stiffly accurate,

$$\begin{gathered} \gamma = \tfrac{1}{2} \ , \quad A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix} , \quad C = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -4/3 & 0 & 0 & 0 \\ -10/3 & -2 & 0 & 0 \\ -1/2 & 0 & 3/2 & 0 \end{bmatrix} , \\ \alpha = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} , \quad g = \begin{bmatrix} 1/2 \\ 1/6 \\ -1/2 \\ 0 \end{bmatrix} , \quad m = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 1 \end{bmatrix} , \quad \hat{m} = \begin{bmatrix} 8/3 \\ 1 \\ 1 \\ -1/3 \end{bmatrix} . \end{gathered} \tag{20}$$

It requires 2 function evaluations. It allows inexact Jacobians and the embedded method is order 3. The stability function is

$$R(z) = \frac{1-z}{(1-z/2)^4}$$

and we have $R(z), R'(z), R''(z), R'''(z) \geq 0$ for $z \leq 0$.

Appendix B. The test mechanism

Consider the basic stratospheric reaction mechanism (adapted from NASA HSRP/AESA [9])

$$
\begin{array}{rcccccccl}
r1) & O_2 & + & h\nu & \xrightarrow{k_1} & 2\,O & & & \left(k_1 = 2.643 \times 10^{-10} \cdot \sigma^3\right) \\
r2) & O & + & O_2 & \xrightarrow{k_2} & O_3 & & & \left(k_2 = 8.018 \times 10^{-17}\right) \\
r3) & O_3 & + & h\nu & \xrightarrow{k_3} & O & + & O_2 & \left(k_3 = 6.120 \times 10^{-04} \cdot \sigma\right) \\
r4) & O & + & O_3 & \xrightarrow{k_4} & 2\,O_2 & & & \left(k_4 = 1.576 \times 10^{-15}\right) \\
r5) & O_3 & + & h\nu & \xrightarrow{k_5} & O^{1D} & + & O_2 & \left(k_5 = 1.070 \times 10^{-03} \cdot \sigma^2\right) \\
r6) & O^{1D} & + & M & \xrightarrow{k_6} & O & + & M & \left(k_6 = 7.110 \times 10^{-11}\right) \\
r7) & O^{1D} & + & O_3 & \xrightarrow{k_7} & 2\,O_2 & & & \left(k_7 = 1.200 \times 10^{-10}\right) \\
r8) & NO & + & O_3 & \xrightarrow{k_8} & NO_2 & + & O_2 & \left(k_8 = 6.062 \times 10^{-15}\right) \\
r9) & NO_2 & + & O & \xrightarrow{k_9} & NO & + & O_2 & \left(k_9 = 1.069 \times 10^{-11}\right) \\
r10) & NO_2 & + & h\nu & \xrightarrow{k_{10}} & NO & + & O & \left(k_{10} = 1.289 \times 10^{-02} \cdot \sigma\right)
\end{array}
\tag{21}
$$

Here $M = 8.120E + 16$ molec/cm^3 is the atmospheric number density; the rate coefficients are scaled for time t in seconds; and $\sigma(t)$ represents the normalized sunlight intensity,

$$
\begin{array}{rcl}
T_L & = & \left(\dfrac{t}{3600}\right) \bmod 24 \;; \quad T_R = 4.5 \text{ (SunRise)}; \quad T_S = 19.5 \text{ (SunSet)}; \\
\sigma(t) & = & \begin{cases} \frac{1}{2} + \frac{1}{2}\cos\left(\pi \left|\frac{2\,T_L - T_R - T_S}{T_S - T_R}\right| \left[\frac{2\,T_L - T_R - T_S}{T_S - T_R}\right]\right) & \text{if } T_R \le T_L \le T_S \\ 0 & \text{otherwise} \end{cases} .
\end{array}
$$

It is easy to see that along any trajectory of the system (21) the number of oxygen and the number of nitrogen atoms are constant,

$$[O^{1D}] + [O] + 3[O_3] + 2[O_2] + [NO] + 2[NO_2] = \text{const} \;, \quad [NO] + [NO_2] = \text{const} \;,$$

therefore if we denote the concentration vector

$$y = \left[\; [O^{1D}],\; [O],\; [O_3],\; [O_2]\; ,[NO]\; ,[NO_2]\; \right]^T \;,$$

the linear equality constraints have the form

$$A^T = \left[\begin{array}{cccccc} 1 & 1 & 3 & 2 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{array}\right] \;, \qquad A^T y(t) = A^T y(t_0) = b \;.$$

The reduced stratospheric system (21) auto-corrects the negative values of O, O^{1D} and NO, which explains the good accuracy of the standard methods. For example, the atomic oxygen destruction term is $-[O]\,(k_2[O_2] + k_4[O_3] + k_9[NO_2])$; since the parenthesis does not depend on any "possibly-negative" concentration, whenever $[O] < 0$ this destruction term is positive (produces O!) and the oxygen concentration increases toward positive values. Not all chemical systems have the auto-correction property. Appending the extra reaction

$$r11) \quad NO \quad + \quad O \quad \xrightarrow{k_{11}} \quad NO_2 \qquad (k_{11} = 1.0E - 8) \tag{22}$$

leads to a non-correcting kinetic scheme as (22) will continue to destroy

References

[1] Ascher, U.M.; Chin, H.; Reich, S.; Stabilization of DAEs and invariant manifolds. *Numerical Mathematics*, 67:131–149, 1994.

System	O^{1D}	O	O_3	O_2	NO	NO_2
(21)	9.906E+01	6.624E+08	5.326E+11	1.697E+16	8.725E+08	2.240E+08

Table 1: Initial concentrations for the simulation (molec/cm^3).

[2] Bolley, C.; Crouzeix, M.; Conservation de la positivite lors de la discretization des problemes d'evolution parabolique. *R.A.I.R.O. Numerical Analysis*, 12(3):237–245, 1978.

[3] Carmichael, G.R.; Peters, L.K.; Kitada, T.; A second generation model for regional-scale transport/ chemistry/ deposition. *Atmospheric Environment* 20:173–188, 1986.

[4] Damian-Iordache, V.; Sandu, A.; Damian-Iordache, M.; Carmichael, G.R.; Potra, F.A.; KPP - A symbolic preprocessor for chemistry kinetics - User's guide. *Technical report*, The University of Iowa, Iowa City, IA 52246, 1995.

[5] Goldfarb, D.; Idnani, A., A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical Programming* 27:1–33, 1983.

[6] Hairer, E.; Norsett, S.P.; Wanner, G.; *Solving Ordinary Differential Equations I. Nonstiff Problems.* Springer-Verlag, Berlin, 1993.

[7] Hairer, E.; Wanner, G.; *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.* Springer-Verlag, Berlin, 1991.

[8] Hundsdorfer, W.; Numerical solution of advection-diffusion-reaction equations. *Technical report* NM-N9603, Department of Numerical Mathematics, CWI, Amsterdam, 1996.

[9] Kinnison, D.E.; NASA HSRP/AESA stratospheric models intercomparison. *NASA ftp site*, contact kinnison1@llnl.gov.

[10] Owren, B.; Simonsen, H.H.; Alternative Integration Methods for Problems in Structural Dynamics. *Computer Methods In Applied Mechanics And Engineering*, 122(1/2):1–10, 1995.

[11] Sandu, A.; Numerical aspects of air quality modeling. *Ph.D. Thesis*, Applied Mathematical and Computational Sciences, The University of Iowa, 1997.

[12] Sandu, A.; Blom, J.G.; Spee, E.; Verwer, J.; Potra, F.A.; Carmichael, G.R.; Benchmarking stiff ODE solvers for atmospheric chemistry equations II - Rosenbrock Solvers. Atmospheric Environment, 31:3459–3472, 1997.

[13] Sandu, A.; Positive numerical integration methods for chemical kinetic systems. Computer Science technical report CSTR-9905, Michigan Technological University, December 1999.

[14] Shampine, L.F.; Conservation laws and the numerical solution of ODEs. *Computers and Mathematics with Applications*, 12B(5/6):1287–1296, 1986.

[15] Verwer, J.G.; Hunsdorfer, W.; Blom, J.G.; Numerical time integration of air pollution models. *Modeling, Analysis and Simulations report*, MAS-R9825, CWI, Amsterdam, 1998.

[16] Verwer, J.; Spee, E.J.; Blom, J.G.; Hunsdorfer, W.; A second order Rosenbrock method applied to photochemical dispersion problems. *SIAM Journal of Scientific Computing*, 20:1456–1480, 1999.

[17] Blom, J.G.; Verwer, J.; A comparison of integration methods for atmospheric transport-chemistry problems. *Journal of Computational and Applied Mathematics*, to appear.

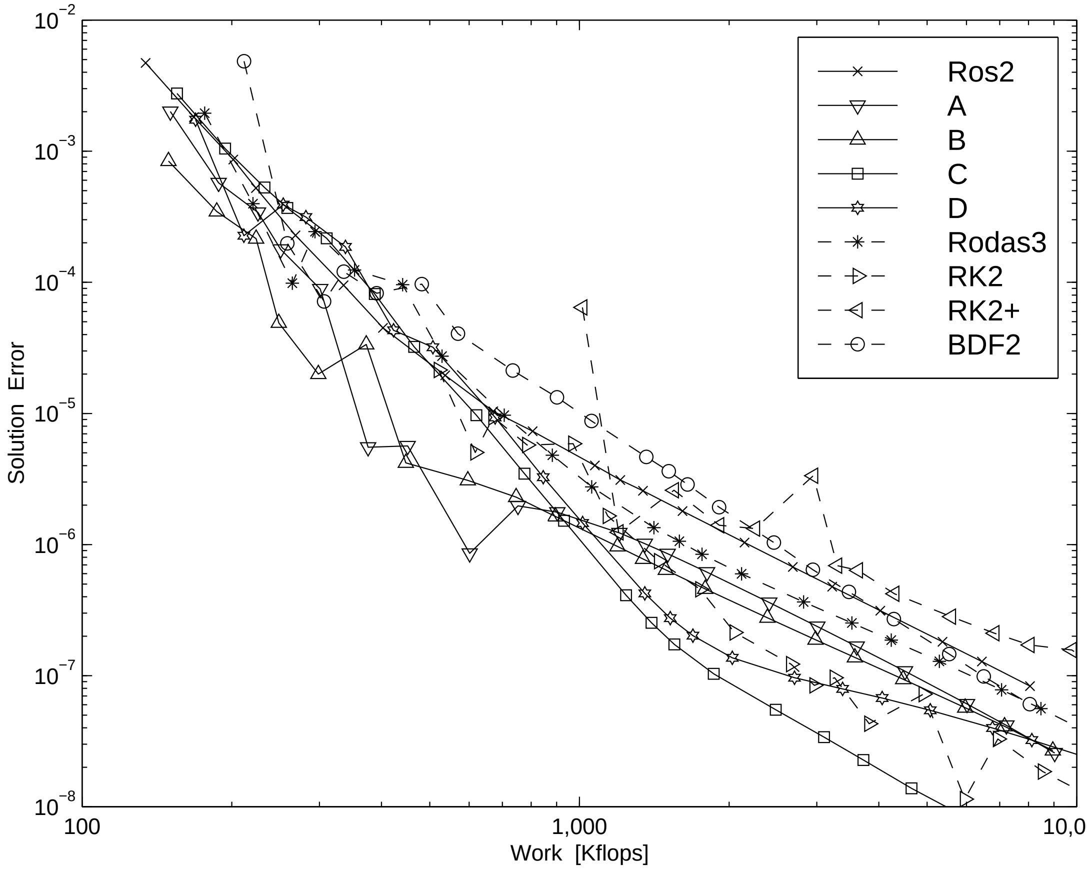

Figure 1: Work-precision diagram shows the solution accuracy versus the computational work. All methods perform similarly.

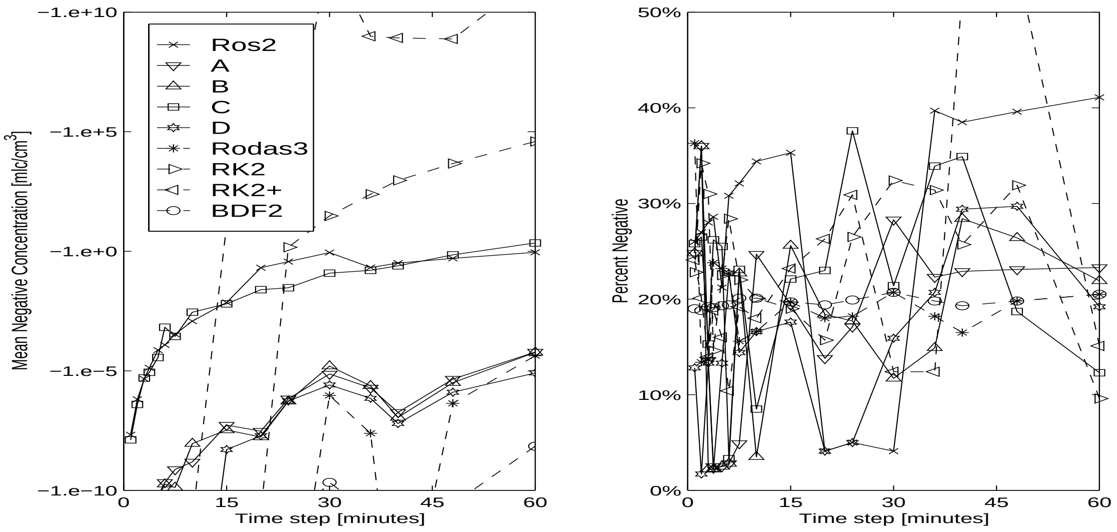

Figure 2: Mean negative values of O^{1D} (left) and the percent of steps which produced negative values (right). Different methods and different step sizes are considered.

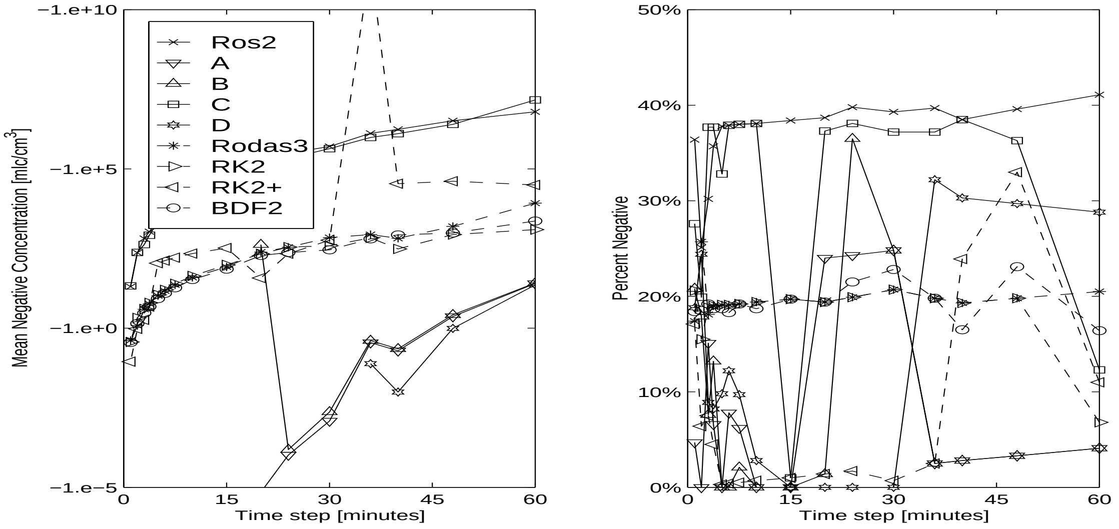

Figure 3: Mean negative values of O (left) and the percent of steps which produced negative values (right). Different methods and different step sizes are considered.

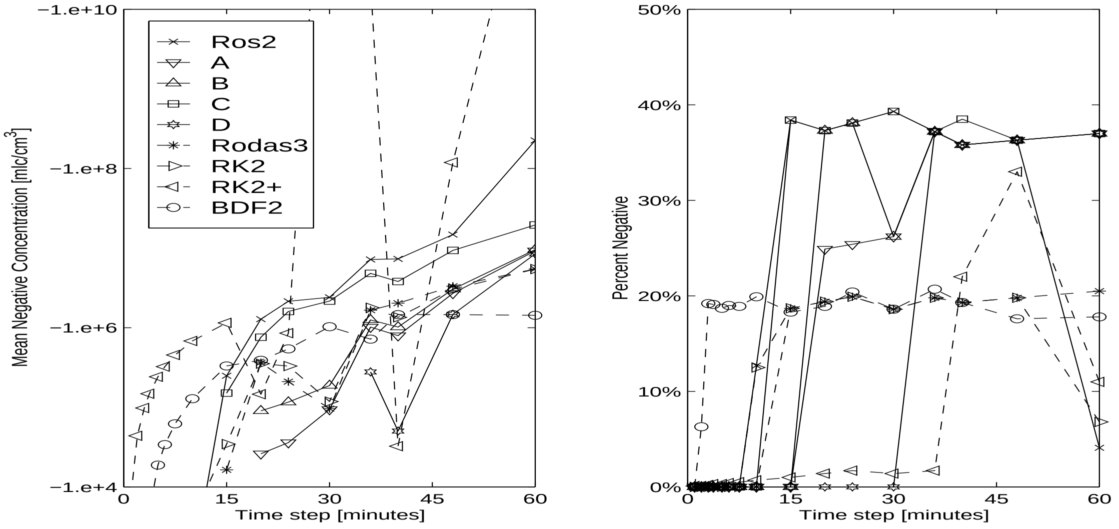

Figure 4: Mean negative values of NO (left) and the percent of steps which produced negative values (right). Different methods and different step sizes are considered.

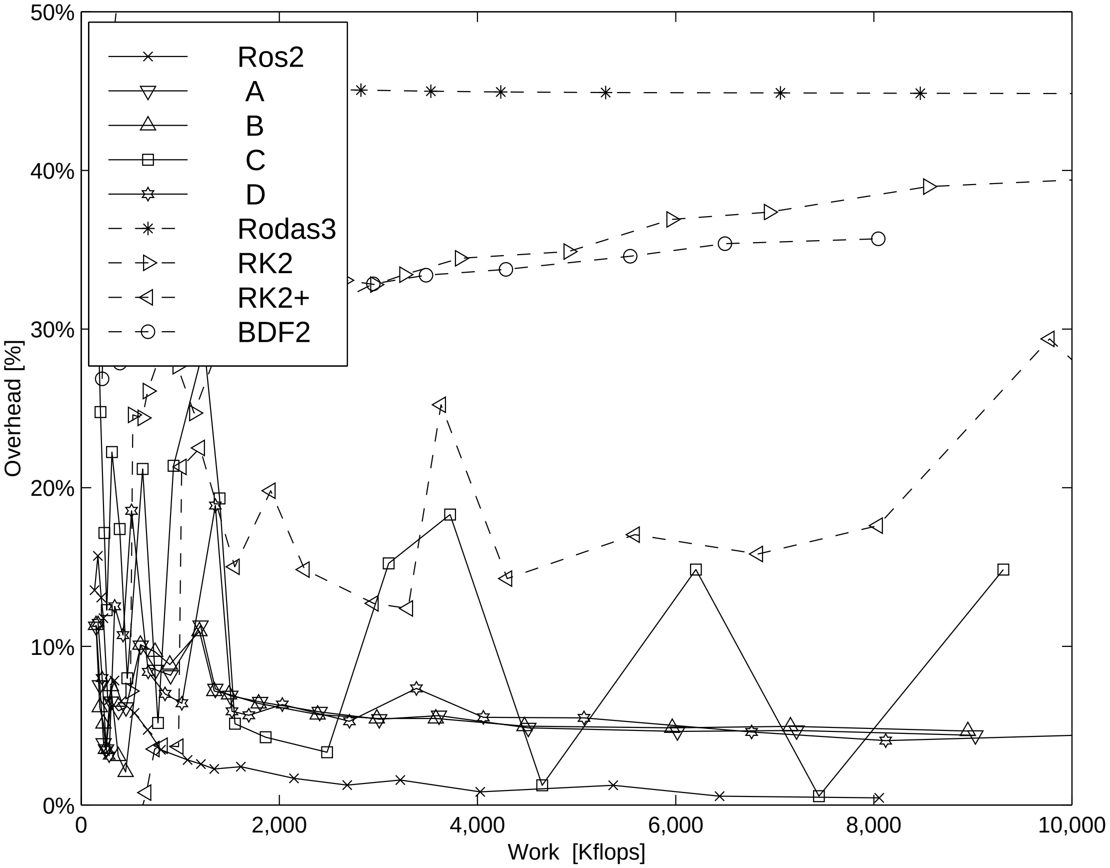

Figure 5: Overheads incurred by solution projection versus computational work of the plain version.