

A Stiff, Variable Time Step Transport Solver for CONTAM

David Lorenzetti¹
W. Stuart Dols²
Andrew Persily²
Michael D. Sohn¹

¹Lawrence Berkeley National Laboratory
1 Cyclotron Road Berkeley, CA 94720

²Engineering Laboratory, National Institute of Standards and Technology
100 Bureau Drive Gaithersburg, MD 20899

Content submitted to and published by:
Building and Environment
Volume 67; pp.260-264

U.S. Department of Commerce
Penny Pritzker, Secretary of Commerce



National Institute of Standards and Technology
Patrick D. Gallagher, Director

DISCLAIMERS

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

Any link(s) to website(s) in this document have been provided because they may have information of interest to our readers. NIST does not necessarily endorse the views expressed or the facts presented on these sites. Further, NIST does not endorse any commercial products that may be advertised or available on these sites.

A stiff, variable time step transport solver for CONTAM

David M. Lorenzetti^{*§} W. Stuart Dols[†] Andrew K. Persily[†] Michael D. Sohn^{*}

12 April 2013

Abstract

We describe the implementation of a new transport solver for CONTAM, a whole-building airflow and contaminant transport model developed by the National Institute of Standards and Technology. Based on CVODE, a general-purpose code for ordinary differential equations, the new solver features variable time steps, high-order integration methods, and automatic error control. These techniques can make CONTAM more accurate when simulating fast transport mechanisms such as high air change rates, sorption, and chemical reactions. We present the relevant theory, then describe the modeling decisions needed to integrate CVODE into CONTAM. Testing with two realistic building models shows that CVODE can run faster than the legacy solvers.

Keywords: contaminant, pollutant, transport, simulation, buildings, CONTAM, CVODE

1 Introduction

CONTAM is a widely-used tool for predicting airflows, contaminant concentrations, and personal exposures in buildings [10]. Applications include designing ventilation and smoke control systems, assessing indoor air quality, and estimating energy use.

CONTAM-3.0 has two contaminant transport solvers. The default solver implements the standard forward and backward Euler methods [2]. The short time step (STS) solver applies forward Euler to transport between rooms, and backward Euler to paths linked in series, such as found in duct systems. This enables STS to model one-dimensional convection-diffusion transport, while avoiding expensive matrix factorizations. Both solvers use fixed time steps, of a length set by the user.

When simulating fast transport mechanisms, such as high air change rates, sorption, and chemical reactions, both solvers may have difficulty accurately predicting transient contaminant concentrations. Because these mechanisms play an important role in determining indoor air quality, we added a third transport solver. The new solver is based on CVODE, a general-purpose code for initial-value problems [7]. CVODE adjusts its time steps to control errors, freeing users from having to select an appropriate time step. In addition, for a given level of accuracy CVODE's higher-order integration methods let it take longer steps, and often run faster, than the default and STS solvers. Finally, the new solver can predict pollutant dynamics on time scales shorter than the one-second limit imposed by the legacy solvers.

After reviewing CONTAM's transport system formulation, we describe CVODE, then discuss the modeling decisions needed to combine the two pieces of software. We also describe some remaining implementation issues. Finally we compare the three solvers using two realistic building models.

^{*}Lawrence Berkeley National Laboratory

[†]National Institute of Standards and Technology

[§]Corresponding author, dmlorenzetti@lbl.gov

2 Theory

CONTAM's transport model treats a building as a collection of nodes, which store contaminant mass, and transport elements, which carry mass between the nodes [3]. The most common type of node, the well-mixed zone, represents contaminant storage in room air, with transport mainly due to airflow. Other node types represent building surfaces, allowing transport by deposition, resuspension, and sorption [10]. A separate solver calculates the airflows, which, with a few exceptions noted below, may be considered fixed over each time step [10].

The transport network defines a system of ordinary differential equations. Let q_i represent the contaminant mass in node i . Then the rate at which mass accumulates in the node, dq_i/dt , equals the net rate of mass transport into the node:

$$\frac{dq_i}{dt} = \sum_e w_i^e, \quad (1)$$

where w_i^e is the rate at which transport element e brings mass into node i . Consider, for example, a flow path that carries a positive mass flow rate $\dot{m}_{i,j}$ of air out of i , and into node j . Then $w_i^e = -\dot{m}_{i,j}x_i$, where x_i is the mass fraction of contaminant in the air in node i . This same element adds mass to node j at a rate $w_j^e = (1 - \eta^e)\dot{m}_{i,j}x_i$, where η^e is the filtration efficiency of the flow path.

In general, the transport rates w_i^e may be nonlinear in the concentrations. However, CONTAM linearizes the system as

$$q' \equiv \frac{dq}{dt} = Kx + e, \quad (2)$$

where q is a vector of contaminant masses in the nodes. The transport rate matrix, K , couples the mass flow rates to the unknown concentrations, x . For instance, an outflow $\dot{m}_{i,j}x_i$ from node i contributes $-\dot{m}_{i,j}$ to the i^{th} diagonal element of K . The excitation vector, e , represents uncoupled transport, for example due to sources, or due to inflow from nodes, such as the outdoors, whose concentration is known. Note that CONTAM uses analytical relations to find K , that is, no derivatives are estimated numerically.

Finally, a diagonal node mass matrix, M , relates the contaminant masses and concentrations:

$$q = Mx. \quad (3)$$

For zones, x_i is a mass fraction, so the i^{th} entry in M is the total mass of air in node i . The default solver finds this node mass using the ideal gas law, with the node pressure taken from the airflow solution.

The default solver discretizes this transport system over the interval from time t to time $t + h$:

$$q_{[t+h]} = q_{[t]} + (1 - \theta)hq'_{[t]} + \theta hq'_{[t+h]}, \quad (4)$$

where $q_{[t]}$ is the numerical estimate of q at time t . The user chooses the time step h , and picks the integration factor $0 \leq \theta \leq 1$. Setting $\theta = 0$ gives the forward Euler method, while $\theta = 1$ gives backward Euler [2]. Note that any choice $\theta > 0$ defines an implicit algebraic system to be solved for $q_{[t+h]}$.

Substituting Eqs. 2 and 3 into Eq. 4 gives the linear system

$$Ax_{[t+h]} = b, \quad (5)$$

where the transport system matrix

$$A = M_{[t+h]} - \theta hK_{[t+h]}, \quad (6)$$

and where the system right-hand-side vector

$$b = (Mx)_{[t]} + (1 - \theta)h(Kx + e)_{[t]} + \theta h e_{[t+h]}. \quad (7)$$

At every time step, the default solver finds $x_{[t+h]}$. For trace contaminants, which by definition do not affect the air density, this takes just one iteration of Eq. 5. Non-trace species, on the other hand, are the constituents of air, so changing their mass fractions affects the node masses in M , and the airflows in K and e . Thus with non-trace contaminants, both A and b depend on x . If the user selects CONTAM’s “Vary density during time step” option, the default solver repeatedly calculates the airflows, updates A and b , and solves Eq. 5 for $x_{[t+h]}$, until the node densities converge.

The user’s choice of h affects the default solver in several ways, most notably its accuracy. Consider a single zone with a source. Defining $\lambda = -M^{-1}K$ and $\sigma = M^{-1}e$, the zone concentration obeys

$$\frac{dx}{dt} = -\lambda x + \sigma. \quad (8)$$

If airflow is the only transport mechanism, λ is the zone’s air change rate [6]. Note σ puts the source strength on a concentration basis.

For λ and σ constant, the exact solution is

$$x_{[t+h]} - \frac{\sigma}{\lambda} = \left(x_{[t]} - \frac{\sigma}{\lambda} \right) e^{-\lambda h}. \quad (9)$$

That is, the concentration should rise or fall to a steady-state value σ/λ , and stepping forward in time by h should reduce the difference $(x - \sigma/\lambda)$ by a factor $e^{-\lambda h}$. However, forward Euler multiplies the difference by $1 - \lambda h$, and backward Euler by $1/(1 + \lambda h)$. Both expressions match the first two terms of the Taylor series expansion of $e^{-\lambda h}$, giving the step a local error on the order of $(\lambda h)^2$ [2]. Thus as λ gets larger, driving the zone to its steady-state concentration faster, the solver requires shorter h in order to maintain the same accuracy. Since halving the step requires twice as many steps to complete a simulation, the global error is proportional to h , and the Euler methods are first-order accurate [8].

For example, with $\lambda h = 0.01$, both methods give a fractional error of about $5 \cdot 10^{-5}$ with each step. For the default solver’s shortest time step, $h = 1$ s, this corresponds to a relatively high air change rate of $\lambda = 36 \text{ h}^{-1}$. Therefore simulating highly-ventilated spaces, or additional loss mechanisms such as deposition or chemical reaction, may exceed the ability of the default solver to control numerical errors at this level.

Long steps also increase the risk of spurious oscillations in the numerical solution. A constant source σ fills the zone of Eq. 8 to a steady-state concentration σ/λ . However, a solver taking steps that are too long may not converge to this value smoothly in time (and may even diverge from it). A stable method finds the correct steady state, even if it makes errors in the transient simulation [2]. The implicit methods defined by setting $\theta > 0$ in the default solver have better stability properties than forward Euler.

A third numerical challenge, stiffness, relates to the presence of multiple time scales in the problem. A stiff system forces a solver to take shorter steps than required to accurately represent the solution [2]. For example, suppose the source in Eq. 8 varies slowly compared to the time scale $1/\lambda$ at which the zone responds to changes in the source. Then, after the initial transients have died, the zone concentration will vary on about the same time scale as the source. Nevertheless, the fast processes represented by λ still determine the appropriate choice of h . Thus stiffness can lead to long solution times, or to roundoff errors associated with taking many short steps. Other sources of stiffness in whole-building transport include chemical reaction and sorption, which may be fast compared to air change rates.

CVODE addresses all these challenges by using higher-order polynomials to approximate the system over time, and by adjusting h to control errors. Where the default solver is structured around the linearized system $q' = Kx + e$, CVODE assumes a general nonlinear problem

$$y' = f\{t, y\}, \quad (10)$$

where y is a vector of state variables, and f is a user-supplied derivative function [7]. CVODE provides two methods for advancing $y\{t\}$ in simulated time: Adams-Moulton (AM), suitable for nonstiff problems; and backward differentiation formula (BDF) for stiff problems.

Adams-Moulton fits a polynomial to the value at time t , plus the derivatives at times $t+h$, t , $t-h$, $t-2h$, and so on:

$$y_{[t+h]} = y_{[t]} + h \sum_{i=0}^p \beta_i y'_{[t-(i-1)h]}. \quad (11)$$

The coefficients β_i depend on the number of past steps considered, $1 \leq p \leq 12$. For example, $p = 1$ gives the trapezoidal method, equivalent to $\theta = 0.5$ in Eq. 4. This does not, however, mean that AM always acts like the default solver with $\theta = 0.5$. Since Adams-Moulton is accurate to order $p + 1$ [8], CVODE seeks to increase p , depending on the simulated history available for constructing the polynomial.

The BDF method also fits a polynomial, but to past values, rather than to past derivatives:

$$y_{[t+h]} = \sum_{i=1}^p \alpha_i y_{[t-(i-1)h]} + h \beta y'_{[t+h]}, \quad (12)$$

where $1 \leq p \leq 5$. Again the coefficients α_i and β depend on p , and again CVODE adjusts p depending on the amount of history available for fitting the polynomial. The BDF method is accurate to order p [8].

In general, the BDF method is stable over a wider range of λh than Adams-Moulton [8]. For both methods, increasing p decreases the region of stability. In practice, however, controlling the step length for accuracy also avoids any stability problems, especially since whole-building contaminant transport problems tend to be stable (i.e., to have $\lambda < 0$ in each node).

Equations 11 and 12 both define algebraic systems to be solved for $y_{[t+h]}$. As with the default solver, CVODE repeatedly linearizes the system into a matrix form like Eq. 5 (although it defines A and b differently). To reduce the number of matrix factorizations, CVODE re-uses the system matrix, A , across time steps, provided the system has not changed too much. Since the cost of factoring a matrix generally varies as the cube of its size [5], this can save significant computational effort.

3 Methods

This section explains the modeling decisions we made in order to integrate CVODE into CONTAM, and improvements that could be made to the current implementation.

Throughout this effort, our first goal was to avoid unintentional changes to existing behavior. To this end, we wrote CONTAM input files to test much of its functionality. This regression test suite comprises 158 input files, plus the expected output, against which we continuously tested the software. Many of these files double as verification tests, comparing the program output to analytical results.

In the most extensive changes to existing code, we rewrote the schedules to synchronize CVODE's time steps. For example, if a contaminant source turns on at time $t = 3$ min, we force CVODE to stop at $t = 3$ min. This includes file-based schedules (for weather, outdoor concentrations, and so on) as well as internal day-schedules (which control fans, contaminant sources, and other modeling elements).

Since most scheduled events change the transport dynamics, we reset CVODE before continuing. Of course, resetting CVODE forces it to take shorter steps, with $p = 1$, until it builds up enough history to switch back to higher-order polynomials. Testing showed that the alternative—letting the solver attempt to extrapolate earlier behavior to the altered system—also induces shorter steps, with the added penalty of initially taking longer steps that the error control mechanism subsequently rejects.

We currently update schedules only between time steps. This introduces errors for schedules that vary continuously with time. Consider a smoothly-varying contaminant source $e\{t\}$ in Eq. 2. In principle, we

should update e every time CVODE evaluates $f\{t, y\}$ in Eq. 10. However, the current implementation fixes the source strength, over each step, at its value from the beginning of the step (since the step length is not known in advance, using the source strength at, e.g., the step midpoint would require re-trying every step).

We took a similar updating approach with airflows. With non-trace contaminants, airflows can vary with the zone concentrations. Therefore a rigorous implementation would re-calculate the airflows on every CVODE evaluation of f . However, we considered this too costly, and simply fix the airflows at their value at the beginning of the step.

To ensure updates take place at reasonable intervals, users can limit CVODE’s longest time step. This also sets the longest period between control element evaluations. A CONTAM control system has sensors, logic components, and actuators, for example to model demand-controlled ventilation [9]. CONTAM updates these control elements between time steps. Since the control system may either respond to, or influence, the contaminant concentrations, limiting the time step can help resolve control actions in time.

Certain control elements need to be evaluated at exact multiples of the user-specified output time step. If a simulation uses reporting, integrating, or averaging controls, we stop CVODE at the requested output times. Otherwise, we allow the solver to step normally, then interpolate its output to the requested times, using the solver’s built-in polynomial fits.

Beyond these timing issues, we found that using CVODE with non-trace contaminants required treating the node masses, q , rather than the mass fractions, x , as the state variables. With non-trace contaminants, the mass fractions in air must always sum to 1. To achieve this, the default solver resets the concentrations of the first non-trace species at the end of every time step. If the first species is some large component of air, such as nitrogen, this does not significantly affect the solution. However, any such change to the system state between time steps, no matter how small, would reset CVODE, preventing it from developing the higher-order polynomial fits it needs to take reasonably long steps. Letting the solver track species masses, rather than mass fractions, avoids this problem.

Finally, where the default solver separates the trace from the non-trace contaminants, the new solver combines them into a single system. This allows chemical reactions between trace and non-trace contaminants, and also avoids any timing issues associated with solving two distinct systems. However, it also produces larger system matrices, with higher factorization costs, for simulations that contain both types of contaminants.

CVODE does not currently support a number of existing CONTAM features, most notably the convection-diffusion model and the computational fluid dynamics (CFD) capability. Nor does it support instantaneous (burst) sources, personal exposures, the sockets interface, nodes with zero volume, or the biconjugate gradient method for iteratively solving transport matrices. All of these features, while technically feasible, would require nontrivial effort to make work under CVODE.

4 Results

To demonstrate CVODE, we compared run times and concentration predictions for two validated building models. The manufactured house model represents a real one-story house [11]. It has 67 rate-determined nodes. A single trace contaminant has a one-minute release at the beginning of the simulation, and a smaller constant source that turns on for four hours in the middle of the day. Airflows are held constant at their initial values. Each run covers a full day of simulated time, writing concentration output once a minute. We tested CVODE’s BDF and AM methods (models MH-BDF and MH-AM) against the default solver with $h = 1$ s and 1 min (MH-DFLT-1s, MH-DFLT-1m), using backward Euler, $\theta = 1$. We also tested the STS solver with $h = 1$ s (MH-STs-1s).

The office building model represents a real 34-story commercial building [4]. In the original model, the air change rate in 16 ventilation-related zones exceeded the stability limit of the STS solver; we replaced

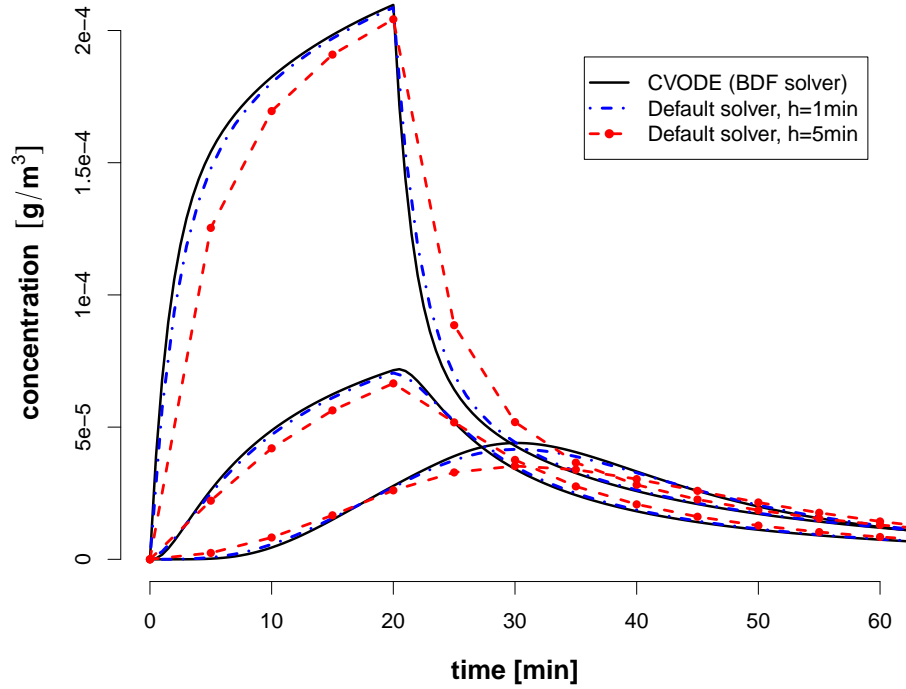


Figure 1: Solver-to-solver differences for three nodes, including the release zone, in the office building. The release zone has $\lambda = 40 \text{ h}^{-1}$, so $h = 1 \text{ min}$ gives $\lambda h \approx 0.67$. Here, $\theta = 1$ in the default solver.

those zones with equivalent ducts. The resulting model has 2809 rate-determined nodes. Two trace contaminants are released for 20 min. The model covers 4 h of simulated time, with output every 5 min. In addition to the options used for the MH model, we tested the default solver with $h = 5 \text{ min}$. Finally, to demonstrate the effect of resetting CVODE, we added a schedule with events at five-minute intervals (OB-BDF-SCH).

Figure 1 shows some solver-to-solver differences for three nodes in the office building. For this model, the default solver with $h = 5 \text{ min}$ introduces significant errors during both the rising and falling transients. Note, however, that over periods when the source stays constant, the error decreases with simulation time, since backward Euler’s stability properties ensure it converges on the correct steady-state concentrations, even if it takes steps too long to predict the transients accurately.

Table 1 summarizes the solver performance. For each combination of solver and model, we timed five runs, discarded the fastest and slowest times, and averaged the remaining three. The table lists the run times, relative to that of the BDF method. In general, the execution speed depends on both how many time steps a solver takes, and how many matrix factorizations it performs. For example, OB-DFLT-5m runs faster than OB-BDF, due to taking fewer time steps; while OB-BDF in turn runs faster than OB-DFLT-1m, due to factoring fewer system matrices.

Comparing model OB-BDF-SCH against OB-BDF shows how scheduled events slow CVODE down, first by cutting steps short, and second by making the solver factor a new system matrix after each event.

The relative run time of the STS solver is much longer on the manufactured house than on the office building. We speculate that this is because the majority of nodes in the manufactured house model represent ductwork, which invokes special handling by the STS solver.

Table 1 also compares concentration predictions. Again taking the BDF method as the reference, at any

Table 1: Comparing the AM, BDF, default, and STS solvers on the manufactured house (MH) and office building (OB) models. For fixed-step solvers, the step length is indicated. Relative run times depend on the compiler used to build the CONTAM executable; the values here (using gcc v4.2.1) are representative. For reference, MH-DFLT-1s takes about 3.4 s to run on a 3 GHz Intel Core 2 processor, while OB-DFLT-1s takes about 4.2 min.

Model	Solver	Solver statistics			NMSE	
		Relative run time	Time steps	Factorizations	Release zone	Max
MH	DFLT-1s	160	86400	86400	$2.8 \text{ e-}9$	$6.0 \text{ e-}9$
	STS-1s	64	86400	0	$4.7 \text{ e-}9$	$7.1 \text{ e-}7$
	DFLT-1m	3.2	1440	1440	$7.9 \text{ e-}6$	$1.5 \text{ e-}5$
	AM	2.4	1191	254	$6.6 \text{ e-}11$	$3.0 \text{ e-}10$
	BDF	1	298	67	–	–
OB	DFLT-1s	71	14400	14400	$9.9 \text{ e-}7$	$4.3 \text{ e-}6$
	AM	6.4	1952	275	$1.8 \text{ e-}9$	$6.6 \text{ e-}9$
	BDF-SCH	6.3	832	340	$2.2 \text{ e-}10$	$2.0 \text{ e-}7$
	STS-1s	4.0	14400	0	$2.3 \text{ e-}4$	$9.6 \text{ e-}3$
	DFLT-1m	1.2	240	240	$3.6 \text{ e-}3$	$1.5 \text{ e-}2$
	BDF	1	288	42	–	–
	DFLT-5m	0.26	48	48	$8.0 \text{ e-}2$	$3.2 \text{ e-}1$

simulation output time t , the difference in a solver’s prediction for node i is

$$d_{i[t]} = x_{i[t]} - x_{i[t]}^{\text{BDF}}. \quad (13)$$

The normalized mean square error (NMSE) for node i aggregates this pointwise difference across the entire simulation [1]:

$$NMSE_i = \frac{\langle d_i^2 \rangle}{\langle x_i \rangle \cdot \langle x_i^{\text{BDF}} \rangle}, \quad (14)$$

where the notation $\langle y \rangle$ indicates the time-averaged value of some quantity y . Note that treating the BDF method as the reference is somewhat arbitrary. Normalizing against the Adams-Moulton solution, or even the default solver with $h = 1$ s, would produce similar results to those in the table.

The table lists the NMSE for the release zone, plus the largest value found among all the nodes. The maximum NMSE occurs outside the release zone because, as the contaminant moves through the building, each node inherits, then adds to, the differences already produced in upstream nodes. Note that, to avoid overstating differences between solvers, the maximum NMSE excludes low-concentration nodes, defined as those with a peak concentration less than a factor 10^{-4} of the peak predicted in the release zone.

For both buildings, CVODE’s error controls keep the AM method, and BDF with schedules, consistent with the BDF solution. By contrast, the default and STS solvers yield a wider range of NMSEs. As expected, shorter time steps give greater agreement with the CVODE solutions. In fact, for both building models the default solver with $h = 1$ s gives results very nearly the same as BDF.

All the solver errors should be judged in the context of modeling a real building. When comparing to experimental data, $NMSE \leq 0.25$ indicates adequate model fit [1]. However this limit allows for measurement errors, parameter uncertainty, and the inability of idealized models to represent the relevant physics [4, 6]. Therefore numerical solutions of a mathematically well-defined model should give much smaller

errors. Consider, for example, the differences shown in Fig. 1 for the office building. In the release zone, the default solver with $h = 5$ min has $NMSE = 0.08$; the worst zone has a normalized error of 0.32. Clearly fixed time steps $h \geq 5$ min are too long for this building. Depending on the application, even one-minute steps may introduce meaningful errors.

For the manufactured house, on the other hand, all the models predict roughly the same concentrations. While MH-DFLT-1m has normalized errors several orders of magnitude greater than the other models, even $NMSE \approx 10^{-5}$ probably is not unreasonable. However, this good agreement in predicted concentrations results partly from the fact that the source remains constant over long stretches of time. For a better indicator that $h = 1$ min is somewhat long for this model, note that the occupied spaces have an average air change rate 13 h^{-1} , making $\lambda h \approx 0.2$.

The results shown in Table 1 use CVODE relative and absolute solution tolerances of 10^{-6} and 10^{-13} , respectively [7]. Loosening these tolerances lets CVODE run faster, but also increases the NMSEs between the various CVODE solutions.

5 Conclusions

The work described here improves CONTAM’s transport predictions in several ways. By adjusting time steps to control the solution error, by employing high-order methods across multiple time steps, and by re-using system matrices when possible, CVODE runs quickly without sacrificing accuracy. Furthermore, unlike the default and STS solvers, CVODE can take sub-one-second time steps when necessary.

For many models, such as the manufactured house, these advantages merely represent a convenience: competitive run times, with no need to perform multiple runs in order to find an appropriate time step. On the other hand, for models with fast transport—for example due to sorption, chemical reactions, or high air change rates—CVODE may be the only solver able to simulate the transport dynamics.

Nevertheless, CVODE is not always the best choice. It does not support all CONTAM features, in particular CFD and convection-diffusion transport. Furthermore the default solver may run faster on models dominated by scheduled events, or when the user chooses to limit CVODE’s longest step. Finally, as suggested by the manufactured house model, long periods of steady-state behavior, especially when combined with fast dynamics, make the simulation results relatively insensitive to the user’s choice of time steps.

When running CVODE on models with continuously-varying schedules, controls, or non-trace contaminants, users should consider limiting CVODE’s longest step. While this will hurt run time, it will ensure updates take place at reasonable intervals. Conversely, removing redundant schedule entries can improve the run time.

When running the default or STS solvers, always check whether reducing the time step changes the predicted concentrations. For a given time step, the default solver with $\theta = 1$ may be more accurate than STS.

The results described here suggest that modifying the default solver, to give it variable time steps, and to make it re-use system matrices, could gain some of the speed enjoyed by CVODE, without sacrificing the advantages that single-step methods bring to event-driven simulation.

Acknowledgments

LBNL’s research was funded in part by the Defense Threat Reduction Agency, and performed under U.S. Department of Energy contract no. DE-AC02-05CH11231. NIST was supported under MIPRs N0017810MP00069 and N0017810MP00160, both funded by the Naval Surface Warfare Center Dahlgren.

References

- [1] *Standard Guide for Statistical Evaluation of Indoor Air Quality Models*. ASTM International, 2008. ASTM D5157-97(2008).
- [2] Uri M. Ascher and Linda R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics, 1998.
- [3] James W. Axley. Multi-zone dispersal analysis by element assembly. *Building and Environment*, 24(2):113–130, 1989.
- [4] Douglas R. Black and Phillip N. Price. Contam airflow models of three large buildings: Model descriptions and validation. Technical Report LBNL-3593E, Lawrence Berkeley National Laboratory, September 2009.
- [5] J.E. Dennis Jr. and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial and Applied Mathematics, 1996.
- [6] Steven J. Emmerich. Validation of multizone IAQ modeling of residential-scale buildings: A review. *ASHRAE Transactions*, 107(2):619–628, 2001.
- [7] Alan C. Hindmarsh and Radu Serban. *User Documentation for CVODE v2.7.0*. Lawrence Livermore National Laboratory, March 2012. UCRL-SM-208108.
- [8] Randall J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. Society for Industrial and Applied Mathematics, 2007.
- [9] Andrew Persily, Amy Musser, Steve Emmerich, and Michael Taylor. Simulations of indoor air quality and ventilation impacts of demand controlled ventilation in commercial and institutional buildings. Technical Report NISTIR 7042, National Institute of Standards and Technology, August 2003.
- [10] George N. Walton and W. Stuart Dols. *CONTAM User Guide and Program Documentation*. National Institute of Standards and Technology, Gaithersburg MD, December 2010. NISTIR 7251.
- [11] Liangzhu (Leon) Wang and Steven J. Emmerich. Modeling the effects of outdoor gasoline powered generator use on indoor carbon monoxide exposures. Technical Report NIST Technical Note 1637, National Institute of Standards and Technology, August 2009.