



# Rapid adaptive Optimization Model for Atmospheric Chemistry (ROMAC) v1.0

Jiangyong Li<sup>1,2</sup>, Chunlin Zhang<sup>1,2</sup>, Wenlong Zhao<sup>1</sup>, Shijie Han<sup>1,2</sup>, Yu Wang<sup>1,3</sup>, Hao Wang<sup>1,2\*</sup>, Boguang Wang<sup>1,2\*</sup>

Correspondence to: Boguang Wang ([tbongue@jnu.edu.cn](mailto:tbongue@jnu.edu.cn)) Hao Wang ([wanghao@jnu.edu.cn](mailto:wanghao@jnu.edu.cn))

5 <sup>1</sup> Australia-China Centre for Air Quality Science and Management (Guangdong), Institute for Environmental and Climate Research, Jinan University, Guangzhou, 511443, China

<sup>2</sup> Guangdong Provincial Observation and Research Station for Atmospheric Environment and Carbon Neutrality in Nanling Forests, Guangzhou, 511443, China

10 <sup>3</sup> Air Quality Studies, Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China

**Abstract.** Rapid adaptive Optimization Model for Atmospheric Chemistry (ROMAC) is a flexible and computationally efficient photochemical box model. The unique adaptive dynamic optimization module in ROMAC enables it to dynamically and rapidly estimate the impact of chemical and physical processes on pollutant concentration. ROMAC overcomes the 15 shortcomings of over-simplified physical modules in traditional box models, and its ability to quantify the effects of chemical and physical processes on pollutant concentrations has been confirmed by the chamber and field observation cases. Since a variable step and variable order numerical solver without Jacobian matrix processing was developed, the computational efficiency of ROMAC is significantly improved. Compared with other box models, the computational efficiency of ROMAC is improved by 96%.

20

**Keywords** Atmospheric chemistry, Photochemical box model, Numerical simulation, High computational efficiency, Stiff ODEs

## 1 Introduction

25 Numerical models are effective tools of atmospheric chemistry studies. The 0-dimensional box model has been widely used in previous studies to study the relationship between secondary pollutants and precursors (Decker et al., 2021; Decker et al., 2019; Ling et al., 2017; Wang et al., 2017; He et al., 2019). Box model can be used as a ground Lagrangian trajectories model to study the influence of regional transport of precursors on the formation of secondary pollutants (Cheng et al., 2010; Wang et al., 2019). In addition, the box model is also a powerful tool in environmental chamber studies (Chen et al., 2015; Novelli 30 et al., 2018).



Since the processes of vertical and horizontal transmission are ignored, the simulation speed of the 0-D box model is higher than that of the 3D air quality model. This allows box models to use more comprehensive chemical mechanisms, and focusing on the analysis of chemical processes. However, with the development of atmospheric chemistry mechanism, the number of chemical reactions involved gradually increases, and the simulation of 0-D box model is still a time-consuming process when using chemical mechanisms with a large number of reactions. Chemical transformations can be described by a series of ordinary differential equations (ODEs), and solving the numerical solution of the ODEs is one of the time-consuming tasks of the box model. For example, the Master Chemical Mechanism (MCM v3.3.1) contains about 5900 species (Jenkin et al., 2015), and the size of the Jacobian matrix is close to 5900×5900, which requires a large number of matrix calculations in the process of solving with the implicit solver. Therefore, it is necessary to develop a computationally efficient model for chemical mechanisms.

Several box models have been developed and applied in previous studies, such as AtChem (Sommariva et al., 2020), Chemistry As A Box Model Application (CAABA) (Sander et al., 2011), Framework for 0-D Atmospheric Modeling (F0AM) (Wolfe et al., 2016), PyCHAM (O'meara et al., 2021), JIBox (Huang and Topping, 2021) and PBM-MCM (Wang et al., 2018). Most of these models rely on third-party tools for differential equation solving. For example, AtChem uses the CVODE library to integrate the ODEs of chemical mechanism. F0AM uses ode15s in MATLAB, a variable order and variable time step solver based on Gear's method. FACSIMILE was used to integrate the ODEs in the PBM-MCM model. Several multistep or multistage approaches are commonly used by these chemical solvers, such as ROSENBROCK, BDF, LSODE, GEAR, SMVGEAR, etc. (Verwer et al., 1996; Aro, 1996b; Sandu et al., 1997a; Sandu et al., 1997b). Although these solving tools have good accuracy and stability, the solving process requires a lot of computing resources, which significantly reduces the computational efficiency.

Previous studies have developed several approaches to improve the efficiency of simulation. One way to improve the computational efficiency is to simplify the chemical mechanism, such as SAPRC07 (Carter, 2012) and CB6 (Yarwood, 2010), which are commonly used in 3D air quality models. The MCM mechanism also has a simplified version (<http://cri.york.ac.uk/>), which can improve the computational efficiency. However, the simplified mechanism will lead to bias in the simulation results of radicals (*e.g.*, OH, HO<sub>2</sub>, RO<sub>2</sub>) and secondary pollutant concentration (Ying and Li, 2011; Jimenez, 2003). Another approach is to improve the computational efficiency of differential equation solver, such as using GPU acceleration (Alvanos and Christoudias, 2017) or using quasi-Newton method (Esentürk et al., 2018). These methods can effectively shorten the running time of the program, but still need to consume a lot of memory, CPU or GPU resources when processing the Jacobian matrix. There are also solution methods that do not need to store and update the Jacobian matrix, such as Quasi-steady State Approximation (QSSA), multistep explicit and semi-implicit methods (Mott et al., 2000; R. and Boris, 1977). But these



methods usually do not conserve mass (Cariolle et al., 2017). There are also fully implicit methods that do not need to deal with the Jacobian matrix, such as Euler Backward Iterative (EBI) (Hertel et al., 1993). EBI method is widely used in 3D chemical transport model (e.g., Community Multiscale Air Quality model, Nested Air Quality Prediction Modeling System) because it is computationally efficient. But EBI solver has a large truncation error because it is only first-order accurate. 65 Another stiff ODEs preconditioner method based on Newton linearization, also simplifies the matrix operations during the solution (Aro, 1996b). However, these algorithms may fail to converge when the Jacobian matrix is significantly off-diagonally dominant (Aro, 1996a). Hence, with the increasing of complexity and scale of chemical mechanism systems, it is still a challenge to make these solving algorithms converge stably.

Rapid adaptive Optimization Model for Atmospheric Chemistry (ROMAC) is a computationally efficient photochemical 70 box model. A variable-step and variable-order solver without Jacobian matrix processing is developed for the ROMAC model. Since the ROMAC model is computationally efficient, accurate and stable, users can dynamically optimize the influence of physical processes on pollutant concentration, and overcome the shortcomings of the lack of physical processes in the traditional box models.

## 2 Description of the ROMAC model

75 ROMAC is a 0-D model focused on the simulation of atmospheric chemical kinetics problem. It was developed to provide users with a flexible and efficient computational tool. The core modules of ROMAC were developed in Fortran, and the data pre-processing and post-processing modules were developed in python, which can keep the model running efficiently and provide users with flexible processing tools. In ROMAC, the changes in concentration of a species can mathematically be represented as Eq. (1).

$$\frac{dc}{dt} = \left[\frac{dc}{dt}\right]_{chem} + \left[\frac{dc}{dt}\right]_{emis} + \left[\frac{dc}{dt}\right]_{dry} + \left[\frac{dc}{dt}\right]_{dilu} + \left[\frac{dc}{dt}\right]_{others} \quad (1)$$

80 Where  $\left[\frac{dc}{dt}\right]_{chem}$  represent the changes due to chemical reactions;  $\left[\frac{dc}{dt}\right]_{emis}$  represents the emission rate for the species;  $\left[\frac{dc}{dt}\right]_{dry}$  and  $\left[\frac{dc}{dt}\right]_{dilu}$  represent the dry deposition and dilution, respectively. For dry deposition, ROMAC uses the maximum dry deposition velocity ( $\text{cm s}^{-1}$ ) calculated by Zhang et al (2003) to estimate the dry deposition process of the species, and user can also customize this value. The dry deposition process is added to the model in the form of first-order kinetics, and the kinetic constant is calculated by the dry deposition velocity and the preset boundary layer height (cm). For dilution, similar to 85 other models (Wolfe et al., 2016; Sommariva et al., 2020), ROMAC uses first-order kinetics to calculate the dilution process, and users can customize the constants of the dilution process.



In addition, ROMAC model sets a user-defined term of rate ( $[\frac{dc}{dt}]_{others}$ ), and the user can add additional change rates if needed, such as the gas-wall partitioning in the chamber studies and the external transport in field observations. Users can also define  $[\frac{dc}{dt}]_{others}$  as physical processes (e.g., vertical and horizontal transport), and estimate the contribution of the physical process to the pollutant concentration by the dynamical optimization algorithm. It helps to overcome the shortcomings of the over-simplified physical modules in the traditional box models.

## 2.1 High efficiency solver for atmospheric chemical kinetic equations

Unlike most existing models, ROMAC does not rely on third-party libraries for numerical solving. ROMAC has its own computationally efficient numerical solver. The solver in ROMAC is optimized according to the characteristics of the atmospheric chemical mechanism, and it will be a universal chemical solver.

Chemical mechanism is the core of atmospheric chemical box model. Generally, chemical reaction equations can be described in Eq. (2).



Where  $\alpha$  and  $\beta$  represents stoichiometric number,  $r$  and  $p$  represent reactant and product, respectively. Hence, derivative of species concentration with respect to time can be described as an ODEs system shown in Eq. (3). For specie  $i$ ,  $f_i$  can be calculated by Eq. (4). In Eq. (4),  $P_{i,t}$  and  $L_{i,t}$  denote the chemical generation rate and the loss rate of species  $i$  at time  $t$ , respectively. It is worth to note that the loss rate is related to the concentration of species  $i$ . Therefore, to facilitate the subsequent formula derivation,  $L_{i,t}$  can be described as a multivariate higher-degree equations for the concentration of species  $i$  shown in Eq.(5). Where  $R_{tot}$  represents the number of the reactions related to the loss rate of species  $i$ ;  $\alpha$  is the stoichiometric number, and  $l_{i,t,R}$  is the part of the chemical reaction rate that is not directly related to the concentration of species  $i$ . The computation of the  $f(C_i, t)$  follows the approach in the Fortran code provided by MCM official website (<http://mcm.york.ac.uk/extract.htm>).

$$[\frac{dC_t}{dt}]_{chem} = f(C_t, t) \quad (3)$$

$$f_i(C_{i,t}, t) = P_{i,t} - L_{i,t} \quad (4)$$

$$L_{i,t} = \sum_{R=1}^{R_{tot}} l_{i,t,R} C_{i,t}^{\alpha_R} \quad (5)$$

The lifetime of different species in atmospheric chemical mechanism varies greatly. For example, OH has an atmospheric lifetime of only seconds but O<sub>3</sub> has a lifetime of several days. Therefore, the ODEs system of atmospheric chemical kinetics simulation is extremely stiff, and explicit methods (e.g., explicit Euler method, explicit Runge-Kutta method) are difficult to solve these problems.



In ROMAC, an implicit Euler method was used to solved the ODEs, the iteration formula is given in Eq. (6). Due to its superior numerical stability, this method has been widely used in other atmospheric chemistry models (Esentürk et al., 2018). However, due to the implicit Euler method only has first-order accuracy, it may introduce large truncation errors in the process of integration. Hence, the trapezoidal method iteration formula shown in Eq. (7) is used for integration in a specific situation.

115 Both implicit Euler method and trapezoidal method have the term of  $f(C_{t+1}, t + 1)$  which is unknown at time  $t$  and needs to be solved. Newton–Raphson (NR) scheme is a popular method for solving such implicit equations. Newton–Raphson scheme will be described by Eq. (6) to Eq. (12). Equations (6) and (7) can be expressed in the form of Eq. (8) and Eq. (9), respectively.

$$C_{t+1} = C_t + f(C_{t+1}, t + 1)\Delta t \quad (6)$$

$$C_{t+1} = C_t + \frac{f(C_t, t) + f(C_{t+1}, t + 1)}{2}\Delta t \quad (7)$$

$$g_1(C_{t+1}) = C_{t+1} - C_t - f(C_{t+1}, t + 1)\Delta t = 0 \quad (8)$$

$$g_2(C_{t+1}) = C_{t+1} - C_t - \frac{f(C_t, t) + f(C_{t+1}, t + 1)}{2}\Delta t = 0 \quad (9)$$

120 So, the iteration formula can be expressed in the form of Eq. (10). Where  $\nabla g^{-1}(C_{t+1})$  the inverse matrix of the Jacobian matrix of  $g(C_{t+1})$ . The Jacobian matrix for the implicit Euler method is given in Eq. (11), and the Jacobian matrix for the trapezoidal method is given in Eq. (12). It should be noted that the size of Jacobian matrix and its inverse matrix will increase with the number of species in chemical mechanism increasing. In particular, dealing with explicit chemical mechanisms (*e.g.*, MCM) would consume a lot of computer resources to store the Jacobian matrix and its inverse matrix. In addition, the inverse of a large-scale Jacobian matrix is quite time-consuming.

$$C_{t+1}^{k+1} = C_{t+1}^k - \nabla g^{-1}(C_{t+1})g(C_{t+1}) \quad (10)$$

$$\nabla g_1(C_{t+1}) = \begin{bmatrix} 1 - \frac{\partial f_1(C_{1,t+1})}{\partial C_{1,t+1}}\Delta t & \dots & -\frac{\partial f_1(C_{1,t+1})}{\partial C_{n,t+1}}\Delta t \\ \vdots & \ddots & \vdots \\ -\frac{\partial f_n(C_{n,t+1})}{\partial C_{1,t+1}}\Delta t & \dots & 1 - \frac{\partial f_n(C_{n,t+1})}{\partial C_{n,t+1}}\Delta t \end{bmatrix} \quad (11)$$

$$\nabla g_2(C_{t+1}) = \begin{bmatrix} 1 - \frac{\partial f_1(C_{1,t+1})}{\partial C_{1,t+1}}\Delta t - \frac{\partial L_{1,t+1}}{\partial C_{1,t+1}}\frac{\Delta t}{2} & \dots & -\frac{\partial f_1(C_{1,t+1})}{\partial C_{n,t+1}}\Delta t + \frac{\partial P_{1,t+1}}{\partial C_{n,t+1}}\frac{\Delta t}{2} - \frac{\partial L_{1,t+1}}{\partial C_{n,t+1}}\frac{\Delta t}{2} \\ \vdots & \ddots & \vdots \\ -\frac{\partial f_n(C_{n,t+1})}{\partial C_{1,t+1}}\Delta t + \frac{\partial P_{n,t+1}}{\partial C_{1,t+1}}\frac{\Delta t}{2} - \frac{\partial L_{n,t+1}}{\partial C_{1,t+1}}\frac{\Delta t}{2} & \dots & 1 - \frac{\partial f_n(C_{n,t+1})}{\partial C_{n,t+1}}\Delta t - \frac{\partial L_{n,t+1}}{\partial C_{n,t+1}}\frac{\Delta t}{2} \end{bmatrix} \quad (12)$$

125 A Simplified-Newton (SN) method can effectively reduce the computational complexity of the iterative process of NR method. Traditional SN method substitute the inverse Jacobian matrix obtained in the first iteration for the inverse matrix in the subsequent iterations. Although the traditional SN method can reduce the amount of computation, it still needs to calculate and store the inverse of the Jacobian matrix at each time step. To further improve the computational efficiency, ROMAC uses a Diagonal-Simplified-Newton (DSN) method to solve the implicit equations.



When the  $\Delta t$  in Eq. (11) and Eq. (12) is small enough, the Jacobian matrix of  $g(C_{t+1})$  will be a diagonally dominant matrix  
 130 or a quasi-diagonally dominant matrix. Under these conditions, the inverse matrix of Jacobian can be approximated by Eq.  
 (13). According to the equations associated with the implicit Euler method in Eq. (1) to Eq. (13), the iteration formula for  
 specie  $i$  is shown in Eq. (14). Where  $k$  represents the number of iterative solutions. Previous study has also shown that such  
 approximations are reliable (Aro, 1996a). Similarly, the approximate inverse of the Jacobian matrix for the trapezoidal method  
 and the iterative formulas for the solution can be derived as shown in Eq. (15) and Eq. (16), respectively.

$$\nabla g_1^{-1}(C_{t+1}) \approx \begin{bmatrix} 1 & \dots & 0 \\ 1 - \frac{\partial f_1(C_{1,t+1})}{\partial C_{1,t+1}} \Delta t & & \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{1 - \frac{\partial f_n(C_{n,t+1})}{\partial C_{n,t+1}} \Delta t} \end{bmatrix} \quad (13)$$

$$C_{i,t+1}^{k+1} = \frac{\sum_{R=1}^{R_{tot}} (\alpha_R - 1) l_{t+1,R} C_{i,t+1}^k \alpha_R \Delta t + C_{i,t} + P_{t+1} \Delta t}{1 + \sum_{R=1}^{R_{tot}} \alpha_R l_{t+1,R} C_{i,t+1}^k \alpha_R^{-1} \Delta t} \quad (14)$$

$$\nabla g_2^{-1}(C_{t+1}) \approx \begin{bmatrix} 1 & \dots & 0 \\ 1 - \frac{\partial f_1(C_{1,t+1})}{\partial C_{1,t+1}} \Delta t - \frac{\partial L_{t+1}}{\partial C_{1,t+1}} \frac{\Delta t}{2} & & \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{1 - \frac{\partial f_n(C_{n,t+1})}{\partial C_{n,t+1}} \Delta t - \frac{\partial L_{t+1}}{\partial C_{n,t+1}} \frac{\Delta t}{2}} \end{bmatrix} \quad (15)$$

$$C_{i,t+1}^{k+1} = \frac{\sum_{R=1}^{R_{tot}} (\alpha_R - 1) l_{t+1,R} C_{i,t+1}^k \alpha_R \Delta t + 2C_{i,t} + P_{i,t} \Delta t - L_{i,t} \Delta t + P_{i,t+1} \Delta t}{2 + \sum_{R=1}^{R_{tot}} \alpha l_{t+1,R} C_{i,t+1}^k \alpha_R^{-1} \Delta t} \quad (16)$$

135 It's worth noting that if all of the stoichiometric number ( $\alpha_R$ ) is equal to 1, Eq.(14) is the same as the iteration formula of EBI  
 solver (Hertel et al., 1993) used in CMAQ model. In this study, Eq. (14) provides a generalized form of the EBI iteration  
 formula. Hertel's (1993) study shows that EBI solver has the advantages of high computational efficiency and high accuracy.  
 However, the convergence condition of this method has not been discussed, such as how to choose the optimal integration  
 time step size to make the solution process stable and convergent. If the time step size was too short, the computational  
 140 efficiency will decrease. However, if the time step size is too large, the Jacobian matrix will not be diagonally dominant, it  
 will lead algorithm hard to converge or even not converge. This problem also exists in the EBI algorithm. Especially for such  
 a complex chemical mechanism as MCM, directly using the EBI scheme will have a large risk of causing the algorithm not to  
 converge. In ROMAC, a variable time step and variable order scheme was developed to balance the computational efficiency  
 and accuracy. The variable time step scheme can also maintain the Jacobian matrix as a quasi-diagonally dominant matrix and  
 145 reduce the risk of convergence failure. Hence, the numerical solver in ROMAC model will overcome the shortcoming of EBI  
 solver.

Actually, it is difficult to use a fixed time step to ensure that the Jacobian matrix is always quasi-diagonally dominant. In  
 order to find the optimal time step, a variable time step size scheme is used in our model. First,  $\Delta t_0$  is defined as an extremely



small positive value to ensure that this value is not less than the rounding error of the computer. According to IEEE Std 754-  
 150 2008 (Committee, 2008),  $\Delta t_0$  is defined as  $2.22 \times 10^{-16}$  seconds in ROMAC. Secondly,  $\Delta t_1$  is defined as atmospheric lifetime of  
 the species with the shortest lifetime in the chemical mechanism, as shown in Eq. (17). Third, a strict diagonal dominance  
 matrix requires that the diagonal elements are greater than the sum of the rest of the elements in the same row, as shown in Eq.  
 (18). Hence,  $\Delta t_{2,i}$  is calculated by Eq.(19) to ensure that Eq.(18) holds, and  $\Delta t_2$  is the minimum in the set of  $\Delta t_{2,i}$  shown in  
 Eq.(19). Where  $i$  represents the rows of the Jacobian matrix. Finally, the initial integration time step size is determined by Eq.  
 155 (21).

$$\Delta t_1 = \left[ \frac{1}{L_t} \right]_{min} \quad (17)$$

$$|\nabla g(C_{t+1})_{i,i}| > \sum_{j=1}^n |\nabla g(C_{t+1})_{i,j}| \quad (18)$$

$$\Delta t_{2,i} = \frac{0.9}{\left( \sum_{j=1}^n \left| \frac{\partial f_i(C_{1,t+1})}{\partial C_{j,t+1}} \right| \right)} \quad (19)$$

$$\Delta t_2 = [\Delta t_{2,i}]_{min} \quad (20)$$

$$\Delta t_{init} = \begin{cases} \Delta t_0 & (\Delta t_0 \geq \Delta t_1 \text{ and } \Delta t_0 \geq \Delta t_2) \\ \Delta t_1 & (\Delta t_0 < \Delta t_1 \text{ and } \Delta t_0 < \Delta t_2 \text{ and } \Delta t_2 \geq \Delta t_1) \\ \Delta t_2 & (\Delta t_0 < \Delta t_1 \text{ and } \Delta t_0 < \Delta t_2 \text{ and } \Delta t_2 < \Delta t_1) \end{cases} \quad (21)$$

In order to improve the computational efficiency, the integration time step size should grow while ensuring the accuracy of  
 the solution. When the time step size grows, the local truncation error (*LTE*) should be controlled. In each step ( $\Delta t$ ), ROMAC  
 model uses both single-step and double-step methods for integration, and the calculated results are recorded as  $C_{\Delta t}$  and  $C_{\frac{\Delta t}{2}}$ ,  
 respectively. *LTE* is estimated by the difference between  $C_{\Delta t}$  and  $C_{\frac{\Delta t}{2}}$  ( $LTE = \left| C_{\frac{\Delta t}{2}} - C_{\Delta t} \right|$ ), and the relative error is  
 160 estimated by Eq. (22). This method has been successfully used in previous study (Aro, 1996b).

$$RERR = \left[ \frac{\left| C_{\frac{\Delta t}{2}} - C_{\Delta t} \right|}{1 + C_{\Delta t}} \right]_{min} \quad (22)$$

The model needs to adjust the integration time step according to the tolerance preset by user. This requires inferencing a  
 maximum integration time step based on the preset tolerance. According to the Lagrange remainder of Taylor formula, the  
*RERR* of the integration result can also be expressed as Eq. (23). Where  $s$  is the order of integration accuracy, equal to 1 for  
 the implicit Euler method and equal to 2 for the trapezoidal method. Similarly, the user-specified maximum integral relative  
 165 error can be expressed as Eq. (24). Where  $\Delta t_{max}$  is an estimate of the maximum step size allowed when the preset *rtol* condition  
 is satisfied. In ROMAC, the values of  $\xi_1$  and  $\xi_2$  in Eq. (23) and Eq. (24) are assumed to be approximate ( $\xi_1 \approx \xi_2$ ). According to  
 Eq. (23) and Eq. (24), the maximum integration time step can be estimated by Eq. (25). Finally, the integration time step is



updated according to Eq. (26) and Eq. (27) to make sure that the time step not larger than the maximum time step. In order to avoid a too accurate result make the integration step size grow too large, when  $\Delta t_{opt}$  is greater than  $\Delta t$  by a factor of 10, the time step is only increased by a factor of 10. In general, as the integration step size increases, the number of iterations (N) required by the solver in this study will also increase. Too many iterations will make the computation time-consuming, so the integration time step is not increased when the solver iteration time exceeds 50 ( $N \geq 50$ ).

$$RERR = \frac{R_n(\Delta t)}{1 + C_{\Delta t}} = \frac{f^{(s+1)}(\xi_1)\Delta t^{s+1}}{(1 + C_{\Delta t}) \times (s + 1)!} \quad (23)$$

$$rtol = \frac{R_n(\Delta t_{max})}{1 + C_{\Delta t}} = \frac{f^{(s+1)}(\xi_2)\Delta t_{max}^{s+1}}{(1 + C_{\Delta t}) \times (s + 1)!} \quad (24)$$

$$\Delta t_{max} = \left(\frac{rtol}{RERR}\right)^{\frac{1}{s+1}} \Delta t_t \quad (25)$$

$$\Delta t_{opt} = 0.9\Delta t_{max} \quad (26)$$

$$\Delta t_{t+1} = \begin{cases} \Delta t_{opt} & (\Delta t_{opt} < 10\Delta t_t \text{ and } N < 50) \\ 10\Delta t_t & (\Delta t_{opt} \geq 10\Delta t_t \text{ and } N < 50) \\ \Delta t_t & (N \geq 50) \end{cases} \quad (27)$$

ROMAC model has a strict control on the truncation error of integration according to the relative tolerance (*rtol*) and the absolute tolerance (*atol*) specified by the user. If  $RERR < rtol$  or  $LTE < atol$ , proceed to the next integration time, otherwise the integration time step is halved and re-integrated until the tolerance requirement is satisfied.

Another important question is whether to choose the implicit Euler method or the trapezoidal method for the integration process. Both implicit Euler method and trapezoidal method are stable for stiff ODEs. However, the solution method used in this study requires the Jacobian matrix to be diagonally dominant or quasi-diagonally dominant. Since the initial time step in Eq. (18) and Eq. (19) are derived from the implicit Euler method, the implicit Euler method is used for integral starting. If the algorithm converges quickly ( $N < 50$ ), then the trapezoidal method is used on the next integration time step. When N is greater than 50, the algorithm is switched to implicit Euler method on the next integration time step to improve the computational efficiency.

The solver for ROMAC uses a variable-step and variable-order approach (VSVOR) to solve stiff ODEs problems. Most of the time, the accuracy of VSVOR is second order. The VSVOR solver has comparable computational efficiency with the EBI solver, and the solution accuracy and stability are better.

## 2.2 Adaptive dynamic optimization module and variables constraints

ROMAC can be run under user-specified variable constraints, including but not limited to concentrations of chemical species, photolysis rate, temperature, humidity, pressure and other meteorological conditions. For concentrations of chemical species, ROMAC provides the user with three different constraint schemes.





190 **Scheme 1:**

Different from previous models, ROMAC provides a novel constraint scheme to use the observed data to constrain model run.

Scheme 1 does not directly input the species concentration, but control the  $\left[\frac{dc}{dt}\right]_{others}$  term with adaptive dynamic optimization algorithm. The default value for  $\left[\frac{dc}{dt}\right]_{others}$  is 0, after integration,  $\Delta\left[\frac{dc}{dt}\right]_{others}$  can be estimated by the gap between the observed and simulated values, as detailed in Eq. (28).

$$\Delta\left[\frac{dc}{dt}\right]_{others} = \begin{cases} \frac{C_{obs,n+1} - C_{model,n+1}}{t_{n+1} - t_n} & (|C_{obs,n+1} - C_{model,n+1}| \leq 0.1 \times |C_{model,n+1}|) \\ \frac{0.1 \times C_{model,n+1}}{t_{n+1} - t_n} \times (-1)^u & (|C_{obs,n+1} - C_{model,n+1}| > 0.1 \times |C_{model,n+1}|) \end{cases} \quad (28)$$

195 Where  $C_{obs}$  represents the observations and  $C_{model}$  represents the simulations. In Eq. (28),  $u = 1$  when  $C_{obs,n+1}$  is less than  $C_{model,n+1}$  and  $u = 2$  when  $C_{obs,n+1}$  is greater than  $C_{model,n+1}$ . In complex systems, changing  $\left[\frac{dc}{dt}\right]_{others}$  may also affect other chemical process, so the relationship between  $\left[\frac{dc}{dt}\right]_{others}$  and simulation results may be nonlinear. Therefore, it is difficult to calculate  $\left[\frac{dc}{dt}\right]_{others}$  in a single iteration. It is necessary to estimate by loop iteration until the difference between observation and simulation reaches a preset tolerance. In this study, the difference between observation and simulation is characterized by  
 200 Root Mean Square Error (*RMSE*) shown in Eq. (29).

$$RMSE = \sqrt{(C_{obs,n+1} - C_{model,n+1})^2} \quad (29)$$

The cyclic dynamically optimization process of  $\left[\frac{dc}{dt}\right]_{others}$  is shown in Figure 1. The iterative updating formula of  $\left[\frac{dc}{dt}\right]_{others}$  based on Newton–Raphson method is given in Eq. (30).

$$\left[\frac{dc}{dt}\right]_{others_{m+1}} = \begin{cases} \left[\frac{dc}{dt}\right]_{others_m} + \Delta\left[\frac{dc}{dt}\right]_{others} & (m = 1) \\ \left[\frac{dc}{dt}\right]_{others_m} - RMSE_m \cdot \left[\frac{dRMSE}{d\left[\frac{dc}{dt}\right]_{others}}\right]_m^{-1} & (m > 1) \end{cases} \quad (30)$$

$$\left[\frac{dRMSE}{d\left[\frac{dc}{dt}\right]_{others}}\right]_m = \left[\frac{\Delta RMSE}{\Delta\left[\frac{dc}{dt}\right]_{others}}\right]_m = \frac{RMSE_m - RMSE_{m-1}}{\left[\frac{dc}{dt}\right]_{others_m} - \left[\frac{dc}{dt}\right]_{others_{m-1}}} \quad (31)$$

Where  $m$  is the number of iterations and  $\Delta\left[\frac{dc}{dt}\right]_{others}$  at the first iteration ( $m = 1$ ) are estimated by Eq. (28). When the number of iterations is greater than 1 ( $m > 1$ ), the update equation of  $\Delta\left[\frac{dc}{dt}\right]_{others}$  is developed base on New-Raphson method. The  
 205 *RMSE* is used as the objective function for optimization, and the derivative of *RMSE* with  $\left[\frac{dc}{dt}\right]_{others}$  is estimated by the Difference Method (DM) shown in Eq. (31).



The  $\left[\frac{dc}{dt}\right]_{others}$  can also be optimized in the mode of kinetic equations (e.g.,  $\left[\frac{dc}{dt}\right]_{others} = k_{others} \times C$ ), and then the kinetic constants ( $k_{others}$ ) can be optimized using a similar process shown in Figure 1. ROMAC model provides an option for the user to switch between these two modes. Furthermore, user can also use other algorithms for dynamic optimization, such as

210 Ensemble Kalman Filter (EnKF).

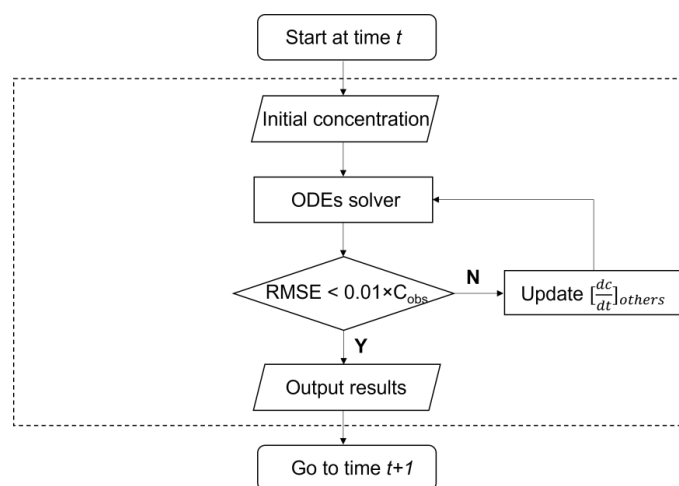


Figure 1. The cyclic dynamically optimization process of  $\left[\frac{dc}{dt}\right]_{others}$ .

#### Scheme 2:

In Scheme 2, the concentration of species can be initialized at the beginning of each simulation time step, which is mainly applied to the solution of initial value problems and more suitable for chamber simulation. This scheme has been widely used

215 in previous models (e.g., PBM-MCM, AtChem, F0AM). However, if the regional transport process of pollutants is not considered, the simulation results of long-lived species in this scheme may have large deviations from the observed results.

#### Scheme 3:

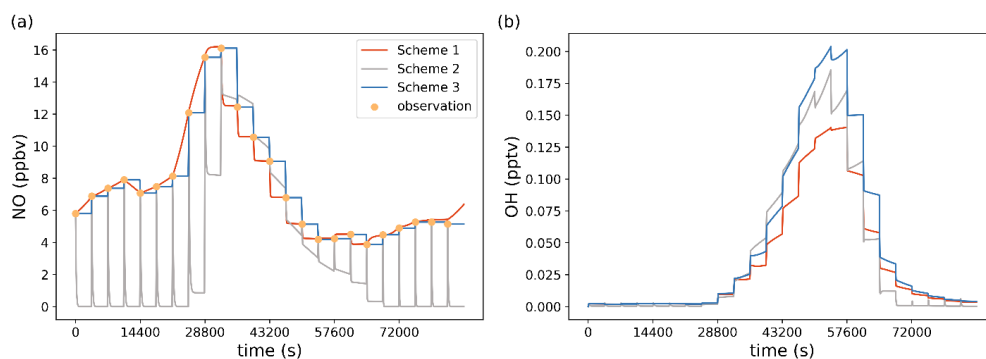
Scheme 3 constrains the change rate of species concentration ( $\frac{dc}{dt} = 0$ ) while constraining the initial concentration, in a similar

220 way as in F0AM. The advantage of this scheme is that the constrained variables can be kept at a user-specified level throughout the simulation. In this scheme, the long-lived species can maintain the observed concentration level. This constraining is appropriate if the temporal resolution of the observed data is high. The time interval of the model should be significantly smaller than the lifetime of constrained species. However, this approach also has its limitations. Since species concentrations are constrained as a constant, chemical imbalances may result in.

225 In order to better understand the characteristics of different constraint schemes, a simple running case was performed. Nitric Oxide was constrained by three different schemes and exhibits its concentration variation characteristics. Figure 2 illustrates the model output results of a test under different schemes to constrain species concentration. The output time step is 120



seconds, and the input interval of the observed data is 3600 seconds. In scheme 2, since emissions and regional transport are not considered, the concentration will decrease rapidly and then reach steady-state. Both scheme1 and scheme 3, the concentration of NO in the model is well consistent with the observed hourly average concentration, which allows the state of the model to be consistent with the real atmospheric state. Due to different constraint schemes, the simulated concentrations of short-lived species (e.g., OH) will also differ. Users should choose a reasonable scheme according to the needs of their research and observation results.



235 **Figure 2. Model output results under different concentration constraint schemes. (a) Variation characteristics of NO concentration in the model under different constraint schemes. (b) OH radical concentrations under different schemes corresponding to the left plot.**

### 2.3 Photolysis

ROMAC provides two ways for the user to set the photolysis rate. First, the user can specify the photolysis rate at each integration time step in the form of an ASCII file. The input photolysis rate can be estimated by other models or the observations. In the ROMAC model, a python script (*TUV2ROMAC.py*) is provided for coupling the output of the Tropospheric Ultraviolet and Visible radiation model (TUVv5.2, available at <https://www2.acom.ucar.edu/modeling/tropospheric-ultraviolet-and-visible-tuv-radiation-model>). Users can easily use this tool to convert the TUV model output results into ROMAC input files.

245 ROMAC provides users with an inline calculation module to calculate photolysis. In the current version, the inline calculation module of photolysis uses the algorithm provided by MCM, an algorithm based on the solar zenith Angle (SZA). The trigonometric parameterization function is shown in Eq. (32). The parameters of  $l$ ,  $m$ ,  $n$  are provided by MCM (<http://mcm.york.ac.uk/>).

$$J = l \times \cos(SZA)^m \times e^{-n \times \sec(SZA)} \quad (32)$$

If both the input photolysis rate and the inline calculated photolysis rate are present, ROMAC will use the input photolysis rate preferentially. In addition, ROMAC provides the user with a photolysis rate modification factor ( $Jrate$ ), users can easily



use this factor to adjust the photolysis rate in the model. The default value of  $J_{rate}$  is 1.0, and the actual photolysis rate used in the model is the input rate or the inline calculated rate multiplied by  $J_{rate}$ .

#### 2.4 Model accuracy and computational efficiency

The comparison of ROMAC with AtChem, F0AM and FACSIMILE which is widely used for MCM was performed on a PC  
255 with a CPU of 16-core AMD Ryzen 9 3950X at 3.5 GHz and 32 GB RAM. The computational efficiency of the model is  
evaluated by CPU time. AtChem, ROMAC, and PBM-MCM are all run using a single core and the CPU time is recorded by  
the software's built-in function. The CPU time used by F0AM is recorded by the function *cputime* in MATLAB. The total  
integration time is 259,200 seconds, and the integration time step is 900 seconds. The settings of *atol* ( $10^{-4}$ ) and *rtol* ( $10^{-3}$ ) in  
the models are consistent. The temperature, pressure and humidity in the scenario simulation are 25°C, 101.325 kPa and 35%,  
260 respectively. The chemical mechanism used in this test is MCM v3.3.1, and the initial species concentrations are shown in  
Table A1. Since running the entire version of MCM v3.3.1 using AtChem is computationally excessive for our computing  
platform, we only selected the VOCs include in EPA's Photochemical Assessment Monitoring Stations (PAMS) Target List  
(<https://www.epa.gov/amtic/>) and exported the mechanism file from MCM website. In this test case, 3,899 species and 11,814  
chemical reactions were included.

265 In this study, we assumed that the solution results of AtChem based on the CVODE library are accurate. Therefore, the  
accuracy of the model is evaluated by calculating the relative difference between the solution results of ROMAC and AtChem  
(Eq. (33)).

$$RE_t = \frac{|C_{ROMAC,t} - C_{AtChem,t}|}{|C_{AtChem,t}|} \times 100\% \quad (33)$$

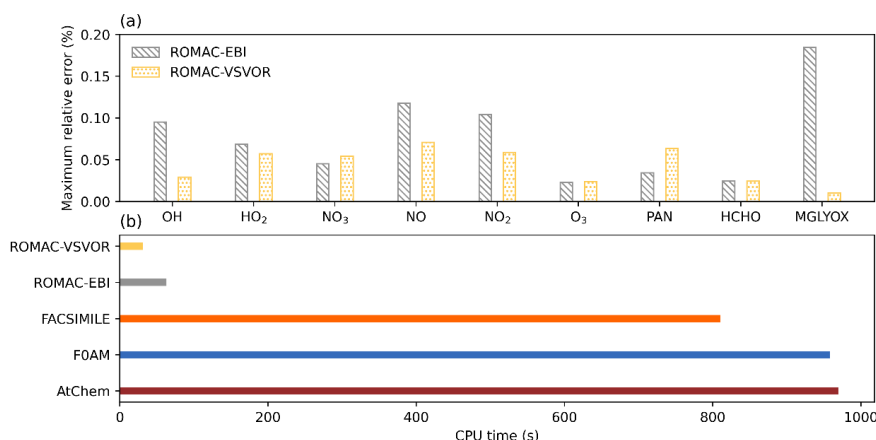
Both EBI and VSVOR solvers in the ROMAC model are evaluated. Figure A1 shows a comparison of simulation results  
for nine species, including radicals and gaseous pollutants, which are commonly used in previous studies to evaluate solution  
270 results (Hertel et al., 1993; Esentürk et al., 2018; Aro, 1996a). As shown in Figure A1, the solution results of ROMAC and  
AtChem are comparable, indicating that the solution results of ROMAC are comparable to the high-precision solution  
algorithm. Figure 3 illustrates the maximum relative error in the scenario simulation and the CPU time used by each model.  
The maximum relative errors between the results of the VSVOR solver and the results of AtChem are all smaller than the  
preset *rtol*. Compared with the single-step EBI solver, the VSVOR solver with variable time step and variable order can better  
275 control the truncation error. Compared with the CPU time required to run, the VSVOR solver with higher solution accuracy is  
even more efficient than EBI. The CPU time consumed by EBI with different integration time steps is shown in Table A2. For  
the MCM chemical mechanism, the algorithm fails to converge when the integration time step is longer than 50 seconds. After



a series of tests, we found that even with an integration time step of 10 seconds, the EBI solver was at risk of failing to converge.

Too small integration time step size makes EBI not efficient in solving MCM mechanism.

280 Compared with other models, ROMAC has greatly improved the computational efficiency of solving large-scale chemical mechanisms. The computational efficiency of ROMAC is 97% higher than that of FOAM and AtChem, and 96% higher than that of FACSIMILE.



285 **Figure 3. Accuracy evaluation and comparison of model computational efficiency. (a) Maximum relative error between the integration results of ROMAC and AtChem. (b) CPU time used to run compare with other models.**

### 3 Application of ROMAC model

#### 3.1 Chamber study

A chamber experiment for toluene degradation was used to evaluate the capabilities of ROMAC model to dynamically optimize chemical and physical processes. In this case, the indoor smog chamber in JNU-VMDSC was used to simulate the degradation of toluene. The JNU-VMDSC provides a reliable experimental platform, and its structure and characterization (e.g., wall loss, light intensity, airtightness test) have been described in previous study (Wang et al., 2023). Toluene and isoprene were injected into the chamber before the UV light was turned on. The initial mixing ratios of toluene and isoprene were 2,157 ppbv and 160 ppbv, respectively.

In order to simulate the effect of the dilution process on the toluene concentration, nitrogen was injected into the chamber at a flow of 7 L/min, while sampling was carried out at a flow of 7 L/min at the sampling port. Similar to previous studies (Dada et al., 2020; Jiang et al., 2020), the rate of dilution was calculated using Eq. (34) and Eq. (35).

$$\left[\frac{dc}{dt}\right]_{dilu} = \frac{C \times dv}{V_{chamber} \times dt} = k_{dilu} \times C \quad (34)$$



$$k_{dilu} = \frac{1}{V_{chamber}} \times \frac{dv}{dt} = \frac{Flow}{V_{chamber}} \quad (35)$$

Where  $k_{dilu}$  is the rate constant of dilution,  $V_{chamber}$  is the volume of chamber (8000 L), and  $Flow$  is the flow of nitrogen injection. Therefore, the theoretically estimation result of  $k_{dilu}$  in this case is  $1.458 \times 10^{-5} \text{ s}^{-1}$ . Wall loss was not considered in this simple experiment with gaseous pollutants.

300 The version of the chemical mechanism used in the model simulations is MCM v3.3.1, all species and mechanisms in MCM are included. Three scenarios case were set up to evaluate the simulation capabilities of ROMAC. In scenario 1, only chemical processes were considered (Eq. (36)). In scenario 2, chemical processes and dilution processes were considered (Eq. (37)). In scenario 3, we assume that the results of the experiment are influenced by an unknown process, and this process is assumed to be a first-order kinetic process (Eq. (38)). The  $k_{others}$  in scenario 3 was dynamically optimized with scheme 1 as described in  
305 Section 2.2. Theoretically, the value of  $k_{others}$  obtained by the dynamically optimization in scenario 3 should be close to  $k_{dilu}$  in scenario 2.

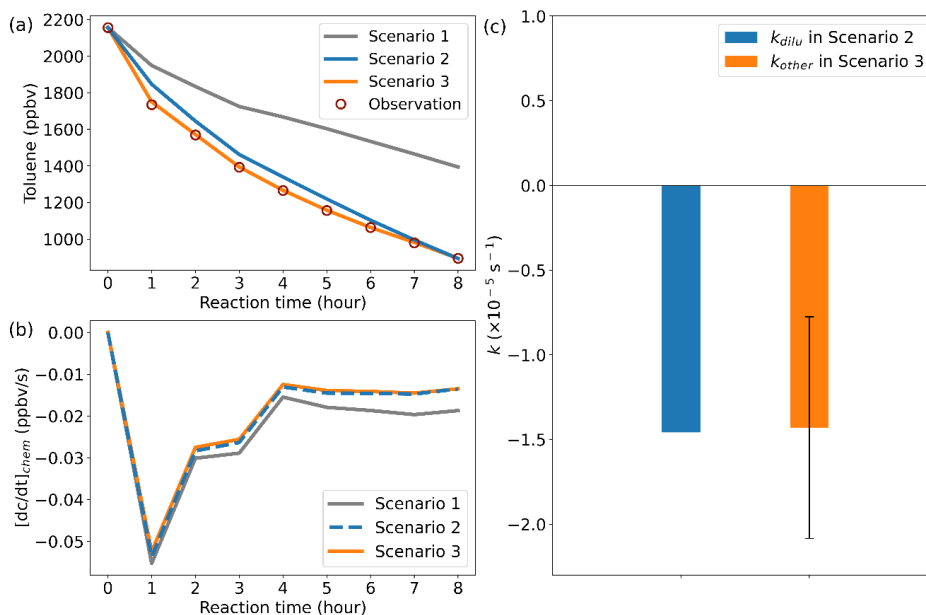
$$\frac{dc_{Tolu}}{dt} = \left[ \frac{dc_{Tolu}}{dt} \right]_{chem} \quad (36)$$

$$\frac{dc_{Tolu}}{dt} = \left[ \frac{dc_{Tolu}}{dt} \right]_{chem} + \left[ \frac{dc_{Tolu}}{dt} \right]_{dilu} = \left[ \frac{dc_{Tolu}}{dt} \right]_{chem} + k_{dilu} \times C_{Tolu} \quad (37)$$

$$\frac{dc_{Tolu}}{dt} = \left[ \frac{dc_{Tolu}}{dt} \right]_{chem} + \left[ \frac{dc_{Tolu}}{dt} \right]_{others} = \left[ \frac{dc_{Tolu}}{dt} \right]_{chem} + k_{others} \times C_{Tolu} \quad (38)$$

The total duration of the chamber experiment was 8 hours, and the CPU time consumed by a single simulation of ROMAC was about 13 seconds. Figure 4 illustrates the comparison results between the simulated and observed toluene mixing ratios for different scenario cases. Due to the lack of dilution process in scenario 1, there is a large gap between simulation results  
310 and observations. After considering the dilution process, the simulation of scenario 2 was improved, which indicates that the setting of scenario 2 is reasonable. The simulation results of scenario 3 agree well with the observations, which indicates that the dynamic optimization algorithm successfully captures the process that cannot be explained by the MCM chemical mechanism. Figure 4b illustrates the chemical loss rate of toluene under different simulation scenarios. The results of scenario 3 and scenario 2 are consistent and significantly different from the results of scenario 1. This indicates that the dynamic  
315 optimization algorithm can improve the chemical process while optimizing the physical process. Ignoring physical processes in the traditional box model may introduce large uncertainty to the simulation results. The optimized value of  $k_{others}$  in scenario 3 is comparable to  $k_{dilu}$  in scenario 2 (Figure 4c), which indicates that the dynamically optimized algorithm is reliable. Based on dynamic optimization, ROMAC can overcome the shortcomings of the over-simplified physical process in the traditional box model.

320



**Figure 4.** Model simulation results. (a) Comparison results between the simulated and observed toluene mixing ratios. (b) Chemical loss rate of toluene. (c) Comparison of kinetic constants in dilution process. Error bars indicate the standard deviation of  $k_{others}$  at different times in scenario 3.

325

### 3.2 Field Observation

This case demonstrates the application of the ROMAC model to the analysis of the photochemical process of  $O_3$  formation and the dynamical optimization of physical processes. The observation data were obtained at the Heshan Atmospheric Supersite (22.728°N, 112.929°E) in Guangdong Province, China. Detailed description of the Heshan site can be found in previous publications (He et al., 2019; Yang et al., 2017). The observation period was from April 4, 2021 to April 10, 2021. Meteorological parameters and the mixing ratios of  $NO_x$ , VOCs,  $SO_2$ , CO were constrained by Scheme 3. The concentrations of  $NO_x$  and VOCs were shown in Figure 5a, with the meteorological observations in Figure A2.

The simulation of  $O_3$  was constrained by Scheme 1. In this case, all physical processes of  $O_3$  (e.g., dry deposition, dilution, transport) were merged into  $[\frac{dc_{O_3}}{dt}]_{others}$ . The rate of change of  $O_3$  is shown in Eq. (39). The optimal estimate of  $[\frac{dc_{O_3}}{dt}]_{others}$  uses the scheme 1 shown in Figure 1.

$$\frac{dc_{O_3}}{dt} = [\frac{dc_{O_3}}{dt}]_{chem} + [\frac{dc_{O_3}}{dt}]_{others} \quad (39)$$

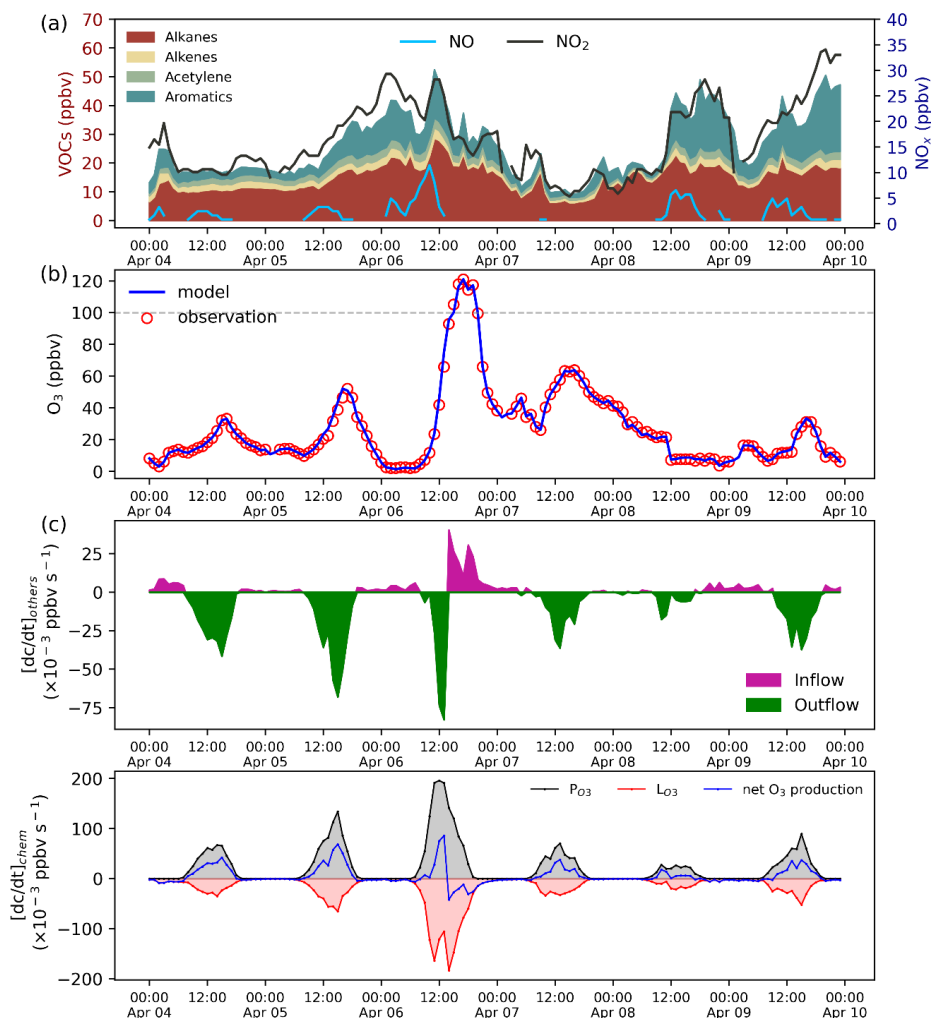


The comparison between the optimized simulation results and the observations of O<sub>3</sub> mixing ratios is shown in Figure 5b. As expected, the model outputs are consistent with the observations due to the dynamic optimization. The estimated value of  $[\frac{dc_{O_3}}{dt}]_{others}$  for the physical process is shown in Figure 5c. Positive values of  $[\frac{dc_{O_3}}{dt}]_{others}$  indicate that physical processes increase local O<sub>3</sub> concentration (*e.g.*, external transport), while negative values indicate that decrease O<sub>3</sub> concentration (*e.g.*, dilution, deposition). As displayed in Figure 5c,  $[\frac{dc_{O_3}}{dt}]_{others}$  is usually negative during the daytime, indicating that O<sub>3</sub> was transported out of the region after formation by photochemical processes. However, positive values of  $[\frac{dc_{O_3}}{dt}]_{others}$  can also occur during the daytime. On April 6, the surface ozone mixing ratio increased rapidly, and the maximum hourly mixing ratio exceeded China II Emission Standard (>100 ppbv). The value of  $[\frac{dc_{O_3}}{dt}]_{others}$  on the afternoon of April 6 is positive, indicating that physical processes were one of the reasons for the occurrence of O<sub>3</sub> pollution.

The rate of O<sub>3</sub> chemical production and precursor sensitivities were calculated using a method described in previous studies (Liu et al., 2022; Wang et al., 2018). As displayed in Figure 5d, the net O<sub>3</sub> production rate on April 6, 2021 was significantly higher than that on other days, indicating that chemical processes were also an important cause of O<sub>3</sub> pollution. The sensitivity of the O<sub>3</sub> formation to its precursors can be represented by relative incremental reactivity (RIR). Figure 6 shows the daily average RIR values of VOCs, NO<sub>x</sub> and CO. The RIR values of VOCs and CO were positive, which indicates that reducing the concentration of VOCs and CO can effectively reduce the chemical formation of O<sub>3</sub>. Except for April 8, the RIR values of NO<sub>x</sub> were negative, indicating that decreasing the NO<sub>x</sub> concentration leads to an increase in O<sub>3</sub> concentration. The negative values of RIR for NO<sub>x</sub> and higher positive values of RIR for VOC indicate that the ozone formation at the Heshan Atmospheric Supersite was mostly likely under VOC limited regime. The result was well consistent with a previous study (He et al., 2019), indicating that the application of ROMAC in chemical process diagnosis is reliable.

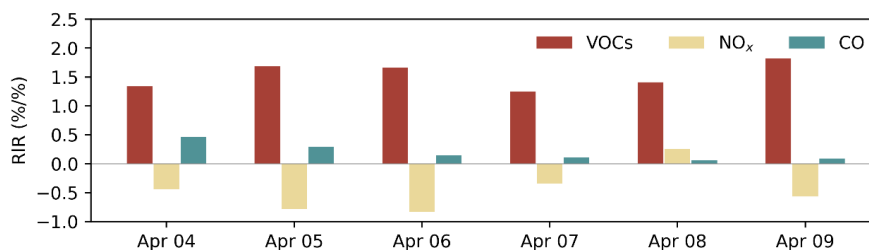
The application of this case demonstrates the ability of the ROMAC model to quantify the contribution of physical and chemical processes to air pollutant concentrations. Compared with the traditional Observation Based box Model (OBM), ROMAC overcomes the shortcomings of over-simplified physical modules. Compared with the emission-based 3D air quality model (*e.g.*, CMAQ, WRF-Chem, NAQPMS), the observation-based dynamic optimization algorithm in ROMAC model reduces the uncertainty introduced by emission inventory and meteorological simulation.





360

**Figure 5.** Species mixing ratio and the rate of O<sub>3</sub> change. (a) VOCs and NO<sub>x</sub> mixing ratios. (b) Model and observation O<sub>3</sub> mixing ratios. (c) The effect of the physical process on the O<sub>3</sub> mixing ratios calculated by the adaptive dynamic optimization module. (d) The rate of O<sub>3</sub> chemical production.



365

**Figure 6.** RIR values of O<sub>3</sub> precursors, *i.e.*, VOCs, NO<sub>x</sub> and CO.



#### 4 Future development

The ROMAC model will be continuously updated and developed. Functionality for future improvements and upgrades,  
370 includes

- A multiphase chemical reaction module and a module for Gas–particle partitioning and sectional simulation are being developed.
- Adjoint sensitivity analysis will be added in a future version, and users can use ROMAC to analyze the relationship between precursors and secondary pollutants.
- 375 — The Ensemble Kalman Filter (EnKF) will be added to dynamically optimize the physical process in future versions.



Appendix A:

Table A1. Initial species concentration used for model comparisons (unit: molecules·cm<sup>-3</sup>)

Species	concentration	Species	concentration	Species	concentration
O3	5.20E+10	NC11H24	4.90E+08	TM123B	1.20E+09
NO2	9.80E+11	NC12H26	9.80E+08	STYRENE	5.90E+09
NO	9.80E+11	C2H4	4.70E+10	C4H6	4.90E+08
CO	1.50E+13	C3H6	5.70E+09	BENZAL	1.90E+10
SO2	7.50E+10	BUT1ENE	7.40E+08	CH3COCH3	9.10E+09
NO3	1.40E+08	TBUT2ENE	2.50E+08	MEK	4.90E+10
C2H2	6.20E+10	C5H8	2.50E+08		
C2H6	9.70E+10	PENT1ENE	1.80E+09		
C3H8	1.40E+11	TPENT2ENE	2.50E+08		
IC4H10	5.00E+10	CPENT2ENE	1.50E+09		
NC4H10	9.90E+10	HEX1ENE	2.50E+08		
IC5H12	1.30E+11	TOLUENE	1.00E+11		
NC5H12	1.70E+11	BENZENE	1.40E+10		
CHEX	1.70E+09	EBENZ	4.20E+10		
M22C4	1.20E+09	OXYL	5.80E+10		
M23C4	7.20E+09	IPBENZ	1.50E+09		
M3PE	7.40E+09	PBENZ	1.20E+09		
NC6H14	8.40E+09	OETHTOL	1.50E+09		
M2HEX	6.60E+09	METHTOL	2.00E+09		
M3HEX	6.40E+09	TM135B	2.50E+09		
NC7H16	4.40E+09	HCHO	1.20E+11		
NC8H18	4.40E+09	CH3CHO	3.90E+10		
NC9H20	4.40E+09	C2H5CHO	3.40E+09		
NC10H22	1.50E+09	C3H7CHO	1.70E+09		
PETHTOL	2.00E+09	MIBK	3.80E+11		
TM124B	2.20E+09	HEX2ONE	6.30E+11		

380 Table A2. CPU time used by the EBI solver at different integration time step sizes (unit: seconds). *Nonconvergence* represents that the EBI solver fails to converge.

Time step	1	10	50	120	900
CPU time	182.8	59.7	<i>Nonconvergence</i>	<i>Nonconvergence</i>	<i>Nonconvergence</i>

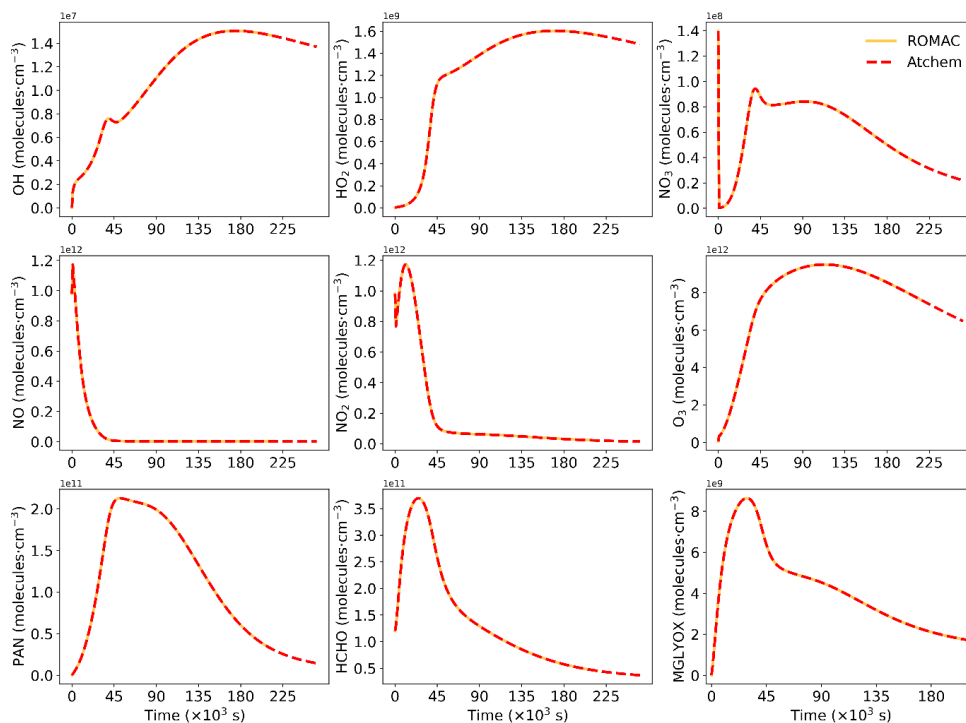
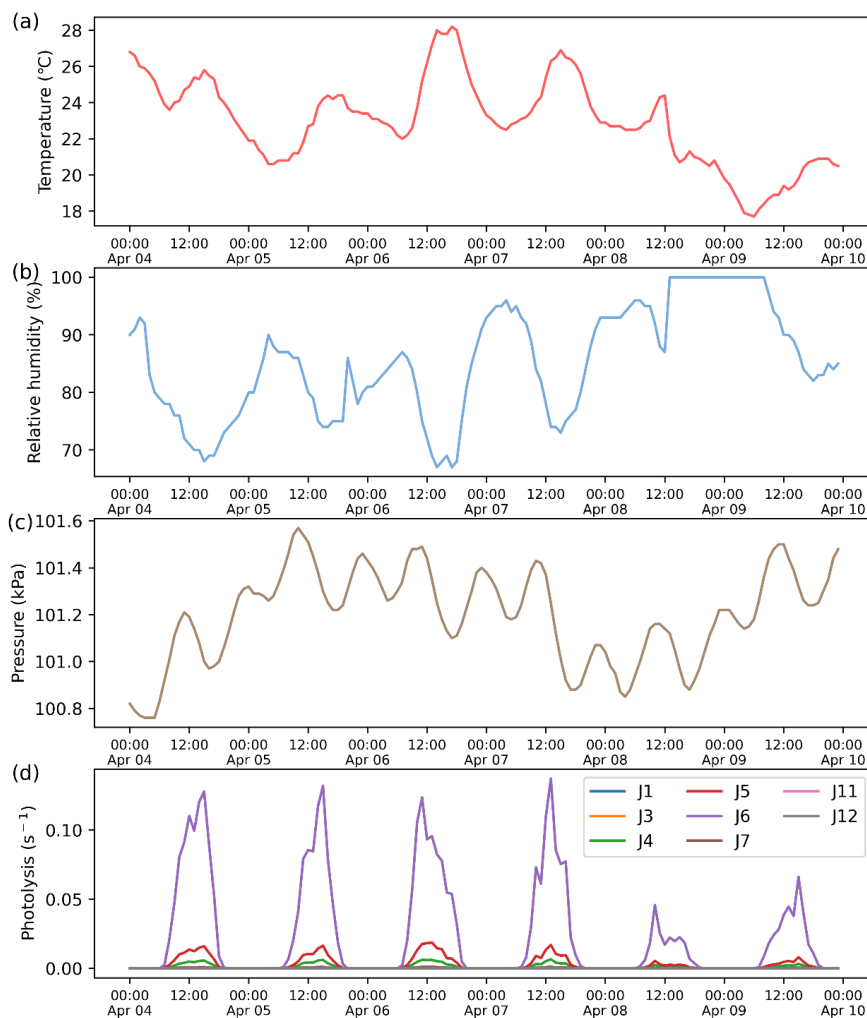


Figure A1. The integration results of the model comparison are compared with AtChem.

385



**Figure A2. Meteorological data input to the model. (a) temperature. (b) relative humidity. (c) atmospheric pressure. (d) photolysis rate**

390 **Code availability**

The current version of ROMAC coupled MCM v3.3.1 is archived on Zenodo: <https://doi.org/10.5281/zenodo.7900781> under the Attribution 4.0 International licence.

**Data availability**

395 The input data used to produce the results used in this paper is archived on Zenodo (<https://doi.org/10.5281/zenodo.7900710>).



#### *Author contributions*

**Jiangyong Li.** The developer of all model source code and algorithms for ROMAC; Conceptualization; Formal analysis;  
400 Writing - Original Draft.

**Chunlin Zhang.** Formal analysis; Writing - Review & Editing.

**Wenlong Zhao.** Formal analysis; Software testing.

**Shijie Han.** The principal investigator of chamber study case; Data curation.

**Yu Wang.** Model Comparison and Evaluation.

405 **Hao Wang.** Funding acquisition; Writing – review & editing.

**Boguang Wang.** Funding acquisition; Writing – review & editing.

#### *Competing interests.*

The authors have declared no competing financial interest.

410

#### *Financial support.*

This work was supported by the National Natural Science Foundation of China (42121004, 42077190), and Science and  
Technology Project of Guangdong Province of China (2019B121202002).

#### 415 *References*

Alvanos, M. and Christoudias, T.: GPU-accelerated atmospheric chemical kinetics in the ECHAM/MESSy (EMAC) Earth  
system model (version 2.52), Geoscientific Model Development, 10, 3679-3693, 10.5194/gmd-10-3679-2017, 2017.

Aro, C. J.: A stiff ODE preconditioner based on Newton linearization, Applied Numerical Mathematic, 21, 335-352, 1996a.

Aro, C. J.: CHEMSODE: a stiff ODE solver for the equations of chemical kinetics, Comput. Phys. Commun, 97, 304-314,  
420 1996b.

Cariolle, D., Moinat, P., Teyssèdre, H., Giraud, L., Josse, B., and Lefèvre, F.: ASIS v1.0: an adaptive solver for the simulation  
of atmospheric chemistry, Geoscientific Model Development, 10, 1467-1485, 10.5194/gmd-10-1467-2017, 2017.

Carter, W. P. L.: SAPRC-07 CHEMICAL MECHANISMS, TEST SIMULATIONS, AND ENVIRONMENTAL CHAMBER  
SIMULATION FILES, 2012.

425 Chen, Y. Z., Sexton, K. G., Jerry, R. E., Surratt, J. D., and Vizuete, W.: Assessment of SAPRC07 with updated isoprene  
chemistry against outdoor chamber experiments, Atmospheric Environment, 105, 109-120, 10.1016/j.atmosenv.2015.01.042,  
2015.



- Cheng, H. R., Guo, H., Saunders, S. M., Lam, S. H. M., Jiang, F., Wang, X. M., Simpson, I. J., Blake, D. R., Louie, P. K. K., and Wang, T. J.: Assessing photochemical ozone formation in the Pearl River Delta with a photochemical trajectory model, *Atmospheric Environment*, 44, 4199-4208, 10.1016/j.atmosenv.2010.07.019, 2010.
- Committee, M. S.: IEEE Standard for Floating-Point Arithmetic, 2008.
- Dada, L., Lehtipalo, K., Kontkanen, J., Nieminen, T., Baalbaki, R., Ahonen, L., Duplissy, J., Yan, C., Chu, B., Petaja, T., Lehtinen, K., Kerminen, V. M., Kulmala, M., and Kangasluoma, J.: Formation and growth of sub-3-nm aerosol particles in experimental chambers, *Nat Protoc*, 15, 1013-1040, 10.1038/s41596-019-0274-z, 2020.
- Decker, Z. C. J., Zarzana, K. J., Coggon, M., Min, K. E., Pollack, I., Ryerson, T. B., Peischl, J., Edwards, P., Dube, W. P., Markovic, M. Z., Roberts, J. M., Veres, P. R., Graus, M., Warneke, C., de Gouw, J., Hatch, L. E., Barsanti, K. C., and Brown, S. S.: Nighttime Chemical Transformation in Biomass Burning Plumes: A Box Model Analysis Initialized with Aircraft Observations, *Environ Sci Technol*, 53, 2529-2538, 10.1021/acs.est.8b05359, 2019.
- Decker, Z. C. J., Robinson, M. A., Barsanti, K. C., Bourgeois, I., Coggon, M. M., DiGangi, J. P., Diskin, G. S., Flocke, F. M., Franchin, A., Fredrickson, C. D., Gkatzelis, G. I., Hall, S. R., Halliday, H., Holmes, C. D., Huey, L. G., Lee, Y. R., Lindaas, J., Middlebrook, A. M., Montzka, D. D., Moore, R., Neuman, J. A., Nowak, J. B., Palm, B. B., Peischl, J., Piel, F., Rickly, P. S., Rollins, A. W., Ryerson, T. B., Schwantes, R. H., Sekimoto, K., Thornhill, L., Thornton, J. A., Tyndall, G. S., Ullmann, K., Van Rooy, P., Veres, P. R., Warneke, C., Washenfelder, R. A., Weinheimer, A. J., Wiggins, E., Winstead, E., Wisthaler, A., Womack, C., and Brown, S. S.: Nighttime and daytime dark oxidation chemistry in wildfire plumes: an observation and model analysis of FIREX-AQ aircraft data, *Atmospheric Chemistry and Physics*, 21, 16293-16317, 10.5194/acp-21-16293-2021, 2021.
- Esentürk, E., Abraham, N. L., Archer-Nicholls, S., Mitsakou, C., Griffiths, P., Archibald, A., and Pyle, J.: Quasi-Newton methods for atmospheric chemistry simulations: implementation in UKCA UM vn10.8, *Geoscientific Model Development*, 11, 3089-3108, 10.5194/gmd-11-3089-2018, 2018.
- He, Z. R., Wang, X. M., Ling, Z. H., Zhao, J., Guo, H., Shao, M., and Wang, Z.: Contributions of different anthropogenic volatile organic compound sources to ozone formation at a receptor site in the Pearl River Delta region and its policy implications, *Atmospheric Chemistry and Physics*, 19, 8801-8816, 10.5194/acp-19-8801-2019, 2019.
- Hertel, O., Berkowicz, R., and Christensen, J.: Test of two numerical schemes for use in atmosphere, *Atmospheric Environment*, 1993.
- Huang, L. and Topping, D.: JIBox v1.1: a Julia-based multi-phase atmospheric chemistry box model, *Geoscientific Model Development*, 14, 2187-2203, 10.5194/gmd-14-2187-2021, 2021.



- Jenkin, M. E., Young, J. C., and Rickard, A. R.: The MCM v3.3.1 degradation scheme for isoprene, *Atmospheric Chemistry and Physics*, 15, 11433-11459, 10.5194/acp-15-11433-2015, 2015.
- Jiang, X., Lv, C., You, B., Liu, Z., Wang, X., and Du, L.: Joint impact of atmospheric SO<sub>2</sub> and NH<sub>3</sub> on the formation of  
460 nanoparticles from photo-oxidation of a typical biomass burning compound, *Environmental Science: Nano*, 7, 2532-2545,  
10.1039/d0en00520g, 2020.
- Jimenez, P.: Comparison of photochemical mechanisms for air quality modeling, *Atmospheric Environment*, 37, 4179-4194,  
10.1016/s1352-2310(03)00567-3, 2003.
- Ling, Z. H., Zhao, J., Fan, S. J., and Wang, X. M.: Sources of formaldehyde and their contributions to photochemical O<sub>3</sub>  
465 formation at an urban site in the Pearl River Delta, southern China, *Chemosphere*, 168, 1293-1301,  
10.1016/j.chemosphere.2016.11.140, 2017.
- Liu, T., Hong, Y., Li, M., Xu, L., Chen, J., Bian, Y., Yang, C., Dan, Y., Zhang, Y., Xue, L., Zhao, M., Huang, Z., and Wang,  
H.: Atmospheric oxidation capacity and ozone pollution mechanism in a coastal city of southeastern China: analysis of a  
typical photochemical episode by an observation-based model, *Atmospheric Chemistry and Physics*, 22, 2173-2190,  
470 10.5194/acp-22-2173-2022, 2022.
- Mott, D. R., Oran, E. S., and van Leer, B.: A Quasi-Steady-State Solver for the Stiff Ordinary Differential Equations of  
Reaction Kinetics, *Journal of Computational Physics*, 164, 407-428, 10.1006/jcph.2000.6605, 2000.
- Novelli, A., Kaminski, M., Rolletter, M., Acir, I. H., Bohn, B., Dorn, H. P., Li, X., Lutz, A., Nehr, S., Rohrer, F., Tillmann,  
R., Wegener, R., Holland, F., Hofzumahaus, A., Kiendler-Scharr, A., Wahner, A., and Fuchs, H.: Evaluation of OH and HO<sub>2</sub>  
475 concentrations and their budgets during photooxidation of 2-methyl-3-butene-2-ol (MBO) in the atmospheric simulation  
chamber SAPHIR, *Atmospheric Chemistry and Physics*, 18, 11409-11422, 10.5194/acp-18-11409-2018, 2018.
- O'Meara, S. P., Xu, S., Topping, D., Alfarra, M. R., Capes, G., Lowe, D., Shao, Y., and McFiggans, G.: PyCHAM (v2.1.1): a  
Python box model for simulating aerosol chambers, *Geoscientific Model Development*, 14, 675-702, 10.5194/gmd-14-675-  
2021, 2021.
- R., Y. T. and Boris, J. P.: A Numerical Technique for Solving Ordinary Differential Equations Associated with the Chemical  
480 Kinetics of Reactive-Flow Problems, *J. Phys. Chem.*, 81, 2424-2427, doi:10.1021/j100540a018, 1977.
- Sander, R., Baumgaertner, A., Gromov, S., Harder, H., Jöckel, P., Kerkweg, A., Kubistin, D., Regelin, E., Riede, H., Sandu,  
A., Taraborrelli, D., Tost, H., and Xie, Z. Q.: The atmospheric chemistry box model CAABA/MECCA-3.0, *Geoscientific  
Model Development*, 4, 373-380, 10.5194/gmd-4-373-2011, 2011.





- 485 Sandu, A., Verwer, J. G., Blom, J. G., Spee, E. J., Carmichael, G. R., and Potra, F. A.: BENCHMARKING STIFF ODE SOLVERS FOR ATMOSPHERIC CHEMISTRY PROBLEMS II: ROSENBROCK SOLVERS *Atmos. Environ.*, 31, 3459-3472, 1997a.
- Sandu, A., Verwer, J. G., Loon, M. V., Carmichael, G. R., Potra, F. A., Dabdub, D., and Seinfeld, J. H.: Benchmarking stiff ode solvers for atmospheric chemistry problems-I. implicit vs explicit, *Atmos. Environ.*, 31, 3151-3166, 1997b.
- 490 Sommariva, R., Cox, S., Martin, C., Borońska, K., Young, J., Jimack, P. K., Pilling, M. J., Matthaïos, V. N., Nelson, B. S., Newland, M. J., Panagi, M., Bloss, W. J., Monks, P. S., and Rickard, A. R.: AtChem (version 1), an open-source box model for the Master Chemical Mechanism, *Geoscientific Model Development*, 13, 169-183, 10.5194/gmd-13-169-2020, 2020.
- Verwer, J. G., Blom, J. G., Loon, M. V., and Spee, E. J.: A comparison of stiff ODE solvers for atmospheric chemistry problems, *Atmos. Environ.*, 30, 49-58, 1996.
- 495 Wang, W., Xiao, Y., Han, S., Zhang, Y., Gong, D., Wang, H., and Wang, B.: A vehicle-mounted dual-smog chamber: Characterization and its preliminary application to evolutionary simulation of photochemical processes in a quasi-realistic atmosphere, *Journal of Environmental Sciences*, 132, 98-108, 10.1016/j.jes.2022.07.034, 2023.
- Wang, Y., Guo, H., Lyu, X., Zhang, L., Zeren, Y., Zou, S., and Ling, Z.: Photochemical evolution of continental air masses and their influence on ozone formation over the South China Sea, *Sci Total Environ.*, 673, 424-434, 10.1016/j.scitotenv.2019.04.075, 2019.
- 500 Wang, Y., Guo, H., Zou, S., Lyu, X., Ling, Z., Cheng, H., and Zeren, Y.: Surface O(3) photochemistry over the South China Sea: Application of a near-explicit chemical mechanism box model, *Environ Pollut.*, 234, 155-166, 10.1016/j.envpol.2017.11.001, 2018.
- Wang, Y., Wang, H., Guo, H., Lyu, X., Cheng, H., Ling, Z., Louie, P. K. K., Simpson, I. J., Meinardi, S., and Blake, D. R.: 505 Long-term O<sub>3</sub>-precursor relationships in Hong Kong: field observation and model simulation, *Atmospheric Chemistry and Physics*, 17, 10919-10935, 10.5194/acp-17-10919-2017, 2017.
- Wolfe, G. M., Marvin, M. R., Roberts, S. J., Travis, K. R., and Liao, J.: The Framework for 0-D Atmospheric Modeling (F0AM) v3.1, *Geoscientific Model Development*, 9, 3309-3319, 10.5194/gmd-9-3309-2016, 2016.
- Yang, Y. D., Shao, M., Kessel, S., Li, Y., Lu, K. D., Lu, S. H., Williams, J., Zhang, Y. H., Zeng, L. M., Noelscher, A. C., Wu, 510 Y. S., Wang, X. M., and Zheng, J. Y.: How the OH reactivity affects the ozone production efficiency: case studies in Beijing and Heshan, China, *Atmospheric Chemistry and Physics*, 17, 7127-7142, 10.5194/acp-17-7127-2017, 2017.
- Yarwood, G.: Development, Evaluation and Testing of Version 6 of the Carbon Bond Chemical Mechanism (CB6), 2010.



Ying, Q. and Li, J.: Implementation and initial application of the near-explicit Master Chemical Mechanism in the 3D Community Multiscale Air Quality (CMAQ) model, *Atmospheric Environment*, 45, 3244-3256, 515 10.1016/j.atmosenv.2011.03.043, 2011.

Zhang, L., Brook, J. R., and Vet, R.: A revised parameterization for gaseous dry deposition in air-quality models, *Atmospheric Chemistry and Physics*, 3, 2067-2082, DOI 10.5194/acp-3-2067-2003, 2003.