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

Jiangyong Li[1,2], Chunlin Zhang[1,2], Wenlong Zhao[1], Shijie Han[1,2], Yu Wang[1,3], Hao Wang[1,2]*, Boguang Wang[1,2]*

Correspondence to: Hao Wang (wanghao@jnu.edu.cn), Boguang Wang (tbongue@jnu.edu.cn)

[1] Australia-China Centre for Air Quality Science and Management (Guangdong), Institute for Environmental and Climate Research, Jinan University, Guangzhou, 511443, China

[2] Guangdong Provincial Observation and Research Station for Atmospheric Environment and Carbon Neutrality in Nanling Forests, Guangzhou, 511443, China

[3] Air Quality Studies, Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China

Abstract. The Rapid adaptive Optimization Model for Atmospheric Chemistry (ROMAC) is a flexible and computationally efficient photochemical box model. Its unique adaptive dynamic optimization module allows for the dynamic and rapid estimation of the impact of chemical and physical processes on pollutant concentration. ROMAC outperforms traditional box models in evaluating the influence of physical processes on pollutant concentrations. Its ability to quantify the effects of chemical and physical processes on pollutant concentrations has been confirmed through chamber and field observation cases. Since the development of a variable-step and variable-order numerical solver that eliminates the need for Jacobian matrix processing, ROMAC's computational efficiency has seen a marked improvement with only a marginal increase in error. Specifically, ROMAC's computational efficiency has improved by 96% when compared to several established box models, such as F0AM and AtChem. Moreover, the solver maintains a discrepancy of less than 0.1% when its results are compared to those obtained from a high-precision solver in AtChem.

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

1 Introduction

Numerical models are effective tools of atmospheric chemistry studies. The 0-dimensional (0-D) box model has been widely used in previous studies to investigate 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, box model is also a powerful tool in environmental chamber studies (Chen et al., 2015; Novelli et al., 2018). 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/MECCA) (Sander et al., 2011; Sander et al., 2019), Framework for 0-D Atmospheric Modeling (F0AM) (Wolfe et al., 2016) , PyCHAM (O'meara et al., 2021), JlBox (Huang and Topping, 2021) and PBM-MCM (Wang 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 3-D air quality model. This allows box models to use more comprehensive chemical mechanisms, and focusing on the analysis of chemical processes. For instance, the utilization of 0-D models in coupling with near-explicit chemical mechanisms can offer a comprehensive comprehension of photochemical processes (Wang et al., 2023b). However, it is important to consider the impact of physical transport on long-lived species, such as its effect on O_3 concentration (Li et al., 2021; Liu et al., 2022). The 0-D model, which lacks a 3-D structure, is unable to directly estimate the impact of physical processes (*e.g.,* vertical and horizontal transport) on pollutant concentrations. Therefore, it is necessary to find a proper scheme to estimate the physical process for these models. Furthermore, as the atmospheric chemistry mechanism continues to develop, the number of chemical reactions involved gradually increases. Consequently, when using a chemical mechanism with massive reactions, the process of obtaining a chemical solution in a 0-D box model also becomes time-consuming. ~~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. Most of box 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, 1996a; Sandu et al., 1997a; Sandu et al., 1997b). Although these third-party solving tools have good accuracy and stability, the solving process requires a lot of computing resources, which significantly reduces the computational efficiency.

The simplified chemical mechanism can effectively improve the solution efficiency of chemical processes, such as SAPRC07 (Carter, 2012), CB6 (Yarwood, 2010), MOZART(Emmons et al., 2010) and the Mainz Organic Mechanism (MOM) (Sander et al., 2019). The MCM mechanism also has a simplified version (http://cri.york.ac.uk/), which can improve the computational efficiency. General methods for reduction (Young & Boris, 1977; Djouad & Sportisse, 2002) and their on-line

implementations (Sander et al., 2019; Shen et al. 2022; Lin et al. 2023) had been developed. These simplified mechanisms generally have a focus on getting particular parts of chemistry. As a result, the simulation results for certain species may diverge from those obtained using near-explicit chemical mechanism, particularly concerning radicals (*e.g.*, OH, HO2, RO2) and the concentrations of secondary pollutants (Ying and Li, 2011; Jimenez, 2003). The adoption of near-explicit chemical mechanisms enables a more detailed representation of the intricate process of photochemical reactions. Consequently, the simplified mechanism cannot adequately replace the role of the near-explicit mechanism. Another approach is to improve the computational efficiency of differential equation solver program, 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 and CPU (or GPU) resources when processing the Jacobian matrix. There are alternative solution methods that do not need to store and update the Jacobian matrix, such 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). In addition, 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.~~ However, EBI solver has a large truncation error because it is only first-order accurate. Another stiff Ordinary Differential Equations (ODEs) preconditioner method based on Newton linearization, also simplifies the matrix operations during the solution (Aro, 1996a). But these algorithms may fail to converge when the Jacobian matrix is significantly off-diagonally dominant (Aro, 1996b). Hence, with the increasing of complexity and scale of chemical mechanism systems, it is still a challenge to make these solving algorithms converge stably.

The Rapid adaptive Optimization Model for Atmospheric Chemistry (ROMAC) is a computationally efficient photochemical box model. To enhance its computational efficiency, a variable-step and variable-order (VSVOR) solver without Jacobian matrix processing was developed for ROMAC. This distinctive solver offers superior computational efficiency in handling atmospheric chemical mechanisms by eliminating the need for third-party libraries for numerical solving. By utilizing the VSVOR solver, ROMAC provides users with the flexibility to dynamically optimize the influence of physical processes on pollutant concentration, which is difficult to achieve in the traditional box model with oversimplified physical modules. Therefore, ROMAC will be computationally efficient and outperform the traditional box models in evaluating the impact of physical processes on pollutant concentrations.

2 Description of the ROMAC model

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).

$$\frac{dc}{dt} = [\frac{dc}{dt}]_{chem} + [\frac{dc}{dt}]_{emis} + [\frac{dc}{dt}]_{dry} + [\frac{dc}{dt}]_{dilu} + [\frac{dc}{dt}]_{others} \tag{1}$$

Where $[\frac{dc}{dt}]_{chem}$ represents the changes due to chemical reactions; $[\frac{dc}{dt}]_{emis}$ represents the emission rate for the species; $[\frac{dc}{dt}]_{dry}$ and $[\frac{dc}{dt}]_{dilu}$ represent the dry deposition and dilution, respectively. For dry deposition, ROMAC uses the maximum dry deposition velocity (cm s^{-1}) calculated by Zhang et al (2003) to estimate the dry deposition process of the species, and users 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). Similar to 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. Note that the current version of ROMAC does not feature a dedicated input function for wet deposition. Instead, the ROMAC model allows users to set a custom rate term, $[\frac{dc}{dt}]_{others}$, which can be employed to account for wet deposition. If wet deposition is important for the simulation case, especially concerning the chemical mechanism of hydrophilic components like sulfate, it is suggested that the user incorporates it into $[\frac{dc}{dt}]_{others}$. Moreover, users have the flexibility to add additional change rates as needed, such as the gas-wall partitioning in the chamber studies or the external transport (*e.g.,* vertical and horizontal transport) in field observations. It is not a difficult task to incorporate new rates of change into $[\frac{dc}{dt}]_{others}$ within the ROMAC framework.

The subsequent sections offer a comprehensive overview of ROMAC's features. Furthermore, to facilitate reference, all the parameters employed in this paper are cataloged in table B1.

2.1 High efficiency solver for atmospheric chemical kinetic equations

Unlike many existing models, ROMAC distinguishes itself by not relying on third-party libraries for numerical solving. Instead, ROMAC employs its own computationally efficient variable-step and variable-order numerical solver named as VSVOR. This solver is engineered to enhance computational efficiency while accommodating the universal attributes of atmospheric chemical mechanism. It approaches all differential equations uniformly, eliminating the need for customized solution schemes

115 solving. VSVOR solver has a control on the truncation error of integration according to the relative tolerance (*rtol*) and the

absolute tolerance (*atol*) specified by the user. The proposed solver offers an algorithm that strikes a balance between efficiency

and accuracy. Most of the time, the accuracy of VSVOR solver can be second order.

Chemical mechanism is the core of atmospheric chemical box model. Generally, chemical reaction equations can be described

in Eq. (2).

$$\alpha_1 r_1 + \alpha_2 r_2 + \ldots + \alpha_n r_n \rightarrow \beta_1 p_1 + \beta_2 p_2 + \ldots + \beta_m p_m \tag{2}$$

120 Where α and β represent 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

125 i shown in Eq.(5). Where R_{tot} represents the total number of the reactions related to the loss rate of species i; α 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 $f(C_t, t)$ follows the approach in the Fortran code provided in the MCM's official website

(http://mcm.york.ac.uk/extract.htt).

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

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

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

The lifetime of different species in atmospheric chemical mechanism varies greatly. For example, OH has an atmospheric

130 lifetime of only seconds but O_3 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) cannot achieve

a stable solution without using a timestep shorter than all lifetimes in the system, which is computationally infeasible.

In ROMAC, implicit Euler method and the trapezoidal method were used to solved the ODEs, the iteration formula is given

in Eq. (6) and Eq. (7), respectively. The implicit Euler method, renowned for its exceptional numerical stability, has found

135 extensive application 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. 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. The Newton–

Raphson (NR) scheme is a widely used method for solving implicit equations in both the implicit Euler method and the trapezoidal method. Equations (6) and (7) can be expressed in the form of Eq. (8) and Eq. (9), respectively.

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

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

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

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

So, the iteration formula of NR scheme can be expressed in the form of Eq. (10). Where $\nabla g^{-1}(C_{t+1})$ is 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 the chemical mechanisms 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.

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

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

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

A Simplified-Newton (SN) method can effectively reduce the computational complexity of the iterative process of NR method. Traditional SN method substitutes 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 Δt in Eq. (11) and Eq. (12) is small enough, the Jacobian matrix of $g(C_{t+1})$ will be a diagonally dominant matrix or a quasi-diagonally dominant matrix. The aforementioned characteristics are inherently present within the Jacobian matrix of chemical mechanisms and are impervious to variations in specific chemical mechanisms. As a result, this scheme proves to be universally applicable across different chemical mechanisms. 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 species 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, 1996b). 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.

160   The solution process was iterated until the difference between the results of two iterations was less than one-tenth of the preset truncation error tolerance (*etc.,* $0.1 \times atol$ or $0.1 \times rtol$) for ODEs solution.

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

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

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

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

It's worth noting that if all of the stoichiometric number (α_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.

165   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 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

170   converge. In ROMAC, the variable time step and variable order scheme helps to balance the computational efficiency and accuracy, maintain the Jacobian matrix as a quasi-diagonally dominant matrix and reduce the risk of convergence failure. Hence, this scheme will enhance the applicability and stability of the ROMAC numerical solver compared to the EBI numerical solver.

    Actually, it is difficult to use a fixed time step to ensure that the Jacobian matrix is always quasi-diagonally dominant. In

175   order to find the optimal time step, a variable time step size scheme is used in our model. First, Δt_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 754-2008 (Committee, 2008), Δt_0 is defined as 2.22×10^{-16} seconds in ROMAC. Secondly, Δt_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 Δt_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. (21).

$$\Delta t_1 = [\frac{1}{L_t}]_{min} \tag{17}$$

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

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

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

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

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 (Δt), 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 estimated by Eq. (22). This method has been successfully used in previous study (Aro, 1996a).

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

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 error can be expressed as Eq. (24). Where Δt_{max} is an estimate of the maximum step size allowed when the preset *rtol* condition is satisfied. In ROMAC, the values of ξ_1 and ξ_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 Δt_{opt} is greater than Δt 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 ($N \geq 50$).

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

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

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

$$\Delta t_{opt} = 0.9\Delta t_{max} \tag{26}$$

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

200   ~~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

205   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 VSVOR solver has comparable computational efficiency with the EBI solver, and the solution accuracy and~~

210   ~~stability are better.~~

~~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.~~

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,

215   photolysis rate, temperature, humidity, pressure and other meteorological conditions. For concentrations of chemical species, ROMAC provides the user with three different constraint schemes.

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 $[\frac{dc}{dt}]_{others}$ term with adaptive dynamic optimization

220   algorithm. The default value for $[\frac{dc}{dt}]_{others}$ is 0, after integration, $\Delta[\frac{dc}{dt}]_{others}$ can be estimated by the gap between the

observed and simulated values, as detailed in Eq. (28).

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

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 $[\frac{dc}{dt}]_{others}$ may also affect

other chemical process, so the relationship between $[\frac{dc}{dt}]_{others}$ and simulation results may be nonlinear. Therefore, it is difficult

225   to calculate $[\frac{dc}{dt}]_{others}$ 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

Root Mean Square Error (*RMSE*) shown in Eq. (29).

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

The cyclic dynamically optimization process of $[\frac{dc}{dt}]_{others}$ is shown in Figure 1. The iterative updating formula of $[\frac{dc}{dt}]_{others}$

based on Newton–Raphson method is given in Eq. (30).

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

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

230   Where m is the number of iterations and $\Delta[\frac{dc}{dt}]_{others}$ 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 $\Delta[\frac{dc}{dt}]_{others}$ is developed base on New-Raphson method. The

RMSE is used as the objective function for optimization, and the derivative of *RMSE* with $[\frac{dc}{dt}]_{others}$ is estimated by the

Difference Method (DM) shown in Eq. (31).

The $[\frac{dc}{dt}]_{others}$ can also be optimized in the mode of kinetic equations (*e.g.*, $[\frac{dc}{dt}]_{others} = k_{others} \times C$), and then the kinetic

235   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 Ensemble Kalman Filter (EnKF).

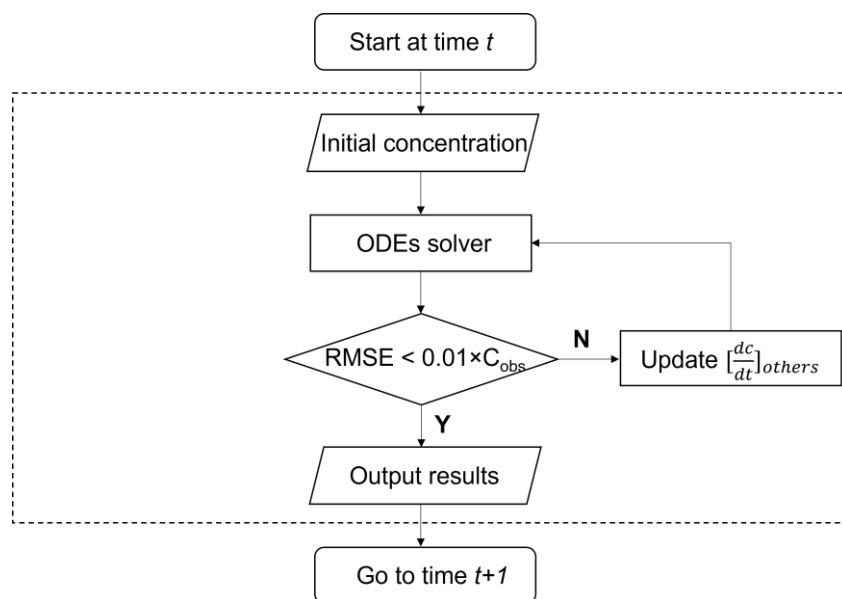



Figure 1. The cyclic dynamically optimization process of $[\frac{dc}{dt}]_{others}$.

Scheme 2:

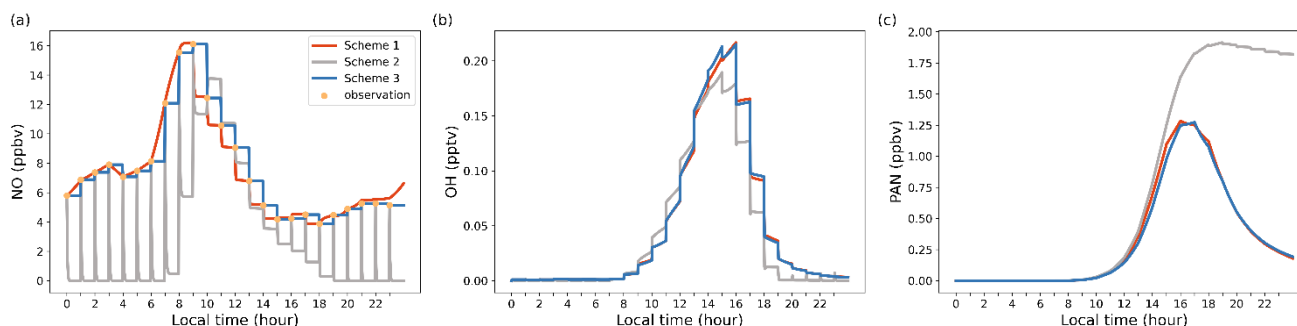
In Scheme 2, the concentration of species can be initialized at the beginning of each simulation time step, which is mainly applied to the solution of initial value problems and more suitable for chamber simulation. This scheme has been widely used in previous models (*e.g.,* PBM-MCM, AtChem, F0AM). 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 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.

To better understand the distinctive attributes of various constraint schemes, we conducted a straightforward test case in which nitric oxide (NO) was constrained by three different schemes. Other input species (*i.e.,* VOCs, NO_2, O_3, SO_2, CO) were constrained by scheme 3. Figure 2 illustrates the results of this test, showcasing how different schemes affect the concentration of target species. The simulation utilized a time step of 120 seconds, while the input data interval from observations was set at 3600 seconds. Under Scheme 2, where emissions and regional transport were not considered, NO concentrations experienced

a rapid decline before reaching a steady-state (Figure 2a). In contrast, both Scheme1 and Scheme 3 displayed NO concentration in close agreement with the observed hourly averages, ensuring that the model faithfully replicated the real atmospheric conditions. It is worth noting that due to variations in constraint schemes, simulated concentrations of other unconstrained

260 species, such as OH and PAN, can also diverge (Figure 2b and 2c). This case study was primarily designed to elucidate the unique features of different constraint schemes, with no intent to definitively validate or invalidate any particular scheme. Users are encouraged to make their scheme selections judiciously, aligning them with their specific research needs and observational findings.

265 **Figure 2. Model output results illustrating diurnal variations for selected species, highlighting the impact of different concentration constraint schemes. (a) NO concentrations; (b) OH concentrations; (c) PAN concentrations.**

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

270 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.

ROMAC provides users with an inline calculation module to calculate photolysis. In the current version, the inline

275 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/).

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

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

280 use this factor to adjust the photolysis rate in the model. The default value of *Jrate* is 1.0, and the actual photolysis rate used in the model is the input rate or the inline calculated rate multiplied by *Jrate*.

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 with a CPU of 16-core AMD Ryzen 9 3950X at 3.5 GHz and 32 GB RAM. The operating system was 64-bit Ubuntu (version

285 20.04.1) and the software was compiled using Intel Fortran (ifort version 2021.2.0). The computational efficiency of the model is evaluated by CPU time. AtChem, ROMAC, and FACSIMILE 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%,

290 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.

295 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)).

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

~~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

300 results (Hertel et al., 1993; Esentürk et al., 2018; Aro, 1996b). The simulation results for ROMAC in Figure A1 are processed by the VSVOR solver. 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. The time series of relative error and its growth rates are depicted in Figure A2. The relative difference between the solution results of ROMAC and that of AtChem gradually stabilizes, and the rate of change of the relative error after 225,000 seconds is extremely low, between $-1.0 \times 10^{-}$

305 ^{6} %/s and 1.0×10^{-6} %/s. This indicates that the error of the ROMAC result was stably controlled. 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*. The solution results

 Compared with the single-step EBI solver, the VSVOR solver with variable time step and variable order can better control the truncation error. In terms of CPU time required for execution , 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. However, reducing the integration time step too much diminishes the efficiency of the EBI solver when handling the MCM mechanism in comparison to the VSVOR solver. Hence, the VSVOR solver exhibits comparable computational efficiency to the EBI solver, while maintaining superior solution accuracy and stability.

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 F0AM and AtChem, and 96% higher than that of FACSIMILE.

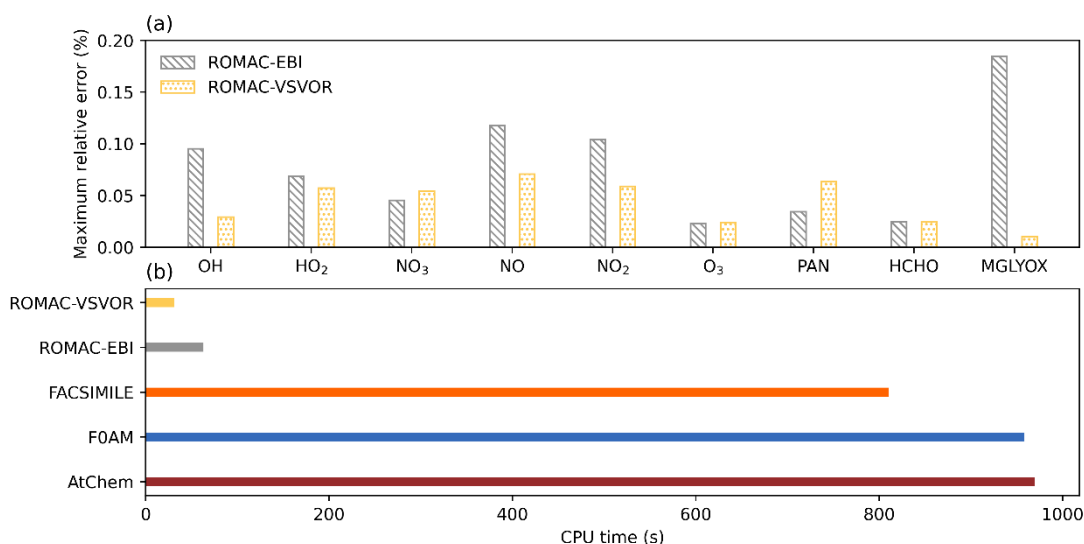



Figure 3. Accuracy evaluation and comparison of model computational efficiency. (a) Maximum relative error between the integration results of ROMAC and AtChem. (b) CPU time used to run compare with other models.

3 Model Validation and Application

3.1 Chamber simulation case

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., 2023a). 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).

$$[\frac{dc}{dt}]_{dilu} = \frac{C \times dv}{V_{chamber} \times dt} = k_{dilu} \times C \tag{34}$$

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

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 1.458×10^{-5} s^{-1}. Wall loss was not considered in this simple experiment with gaseous pollutants.

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 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.

$$\frac{dc_{Tolu}}{dt} = [\frac{dc_{Tolu}}{dt}]_{chem} \tag{36}$$

$$\frac{dc_{Tolu}}{dt} = [\frac{dc_{Tolu}}{dt}]_{chem} + [\frac{dc_{Tolu}}{dt}]_{dilu} = [\frac{dc_{Tolu}}{dt}]_{chem} + k_{dilu} \times C_{Tolu} \tag{37}$$

$$\frac{dc_{Tolu}}{dt} = [\frac{dc_{Tolu}}{dt}]_{chem} + [\frac{dc_{Tolu}}{dt}]_{others} = [\frac{dc_{Tolu}}{dt}]_{chem} + k_{others} \times C_{Tolu} \tag{38}$$

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 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 absence of a physical process in the model is identified as the primary cause for failure to reproduce observations in scenario 1. 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 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 rate of the physical process is subject to uncertainty in practical applications, but its average value is expected to closely approximate the theoretical value. The optimized value of k_{others} in scenario 3, as shown in Figure 4c, exhibits a certain range of fluctuations rather than a fixed value. However, its average values (1.430×10^{-5}) are 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.~~

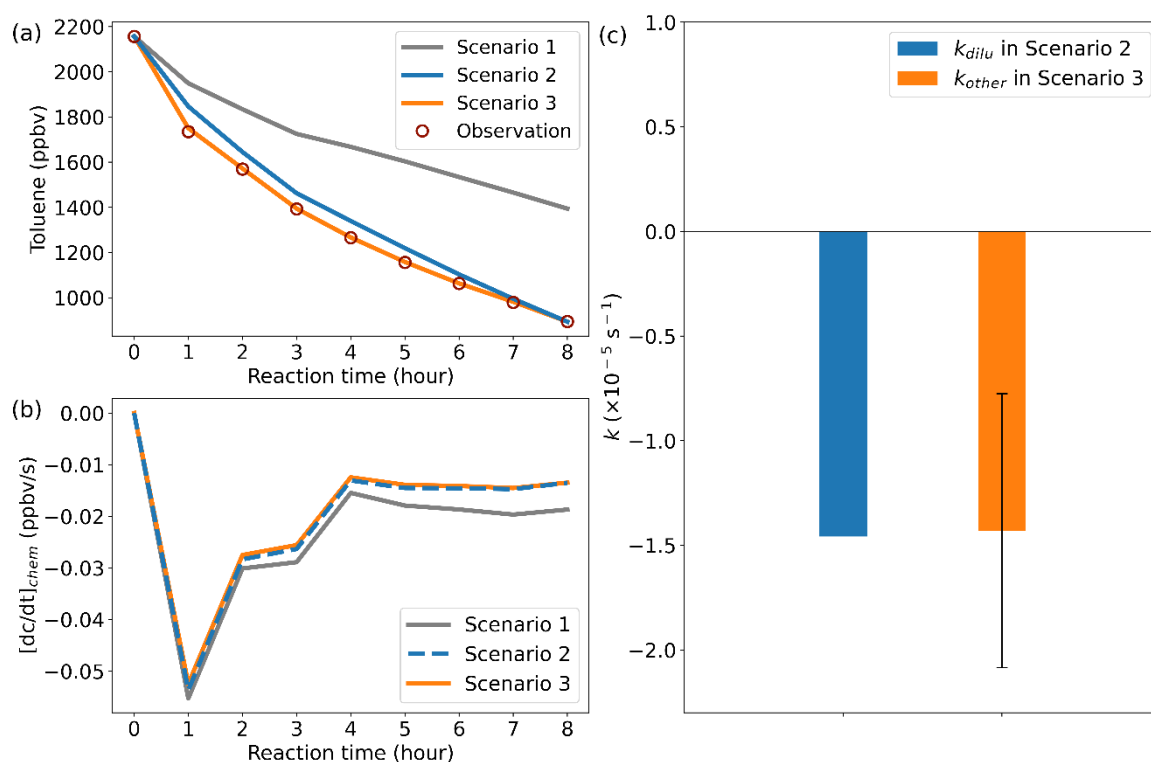



Figure 4. Model simulation results. (a) Comparison results between the simulated and observed toluene mixing ratios. (b) Chemical loss rate of toluene. (c) Comparison of kinetic constants in dilution process. Error bars indicate the standard deviation of k_{others} at different times in scenario 3.

3.2 Field observation case

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

370 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 A3.

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 $[\frac{dc_{O_3}}{dt}]_{others}$

375 uses the scheme 1 shown in Figure 1.

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

The comparison between the optimized simulation results and the observations of O_3 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_3 concentration (*e.g.,* external transport), while negative values indicate that decrease O_3 concentration (*e.g.,*

380 dilution, deposition). As displayed in Figure 5c, $[\frac{dc_{O_3}}{dt}]_{others}$ is usually negative during the daytime, indicating that O_3 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_3 pollution.

385 The rate of O_3 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_3 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_3 pollution. The sensitivity of the O_3 formation to its precursors can be represented by relative incremental reactivity (RIR). Figure 6 shows the daily average RIR values of VOCs, NO_x and CO. The RIR values of VOCs and CO were positive, which indicates that reducing the

390 concentration of VOCs and CO can effectively reduce the chemical formation of O_3. Except for April 8, the RIR values of NO_x were negative, indicating that decreasing the NO_x concentration leads to an increase in O_3 concentration. The negative values of RIR for NOx 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.

395 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 is superior in evaluating the impact of physical processes on pollutant concentrations. 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.

400

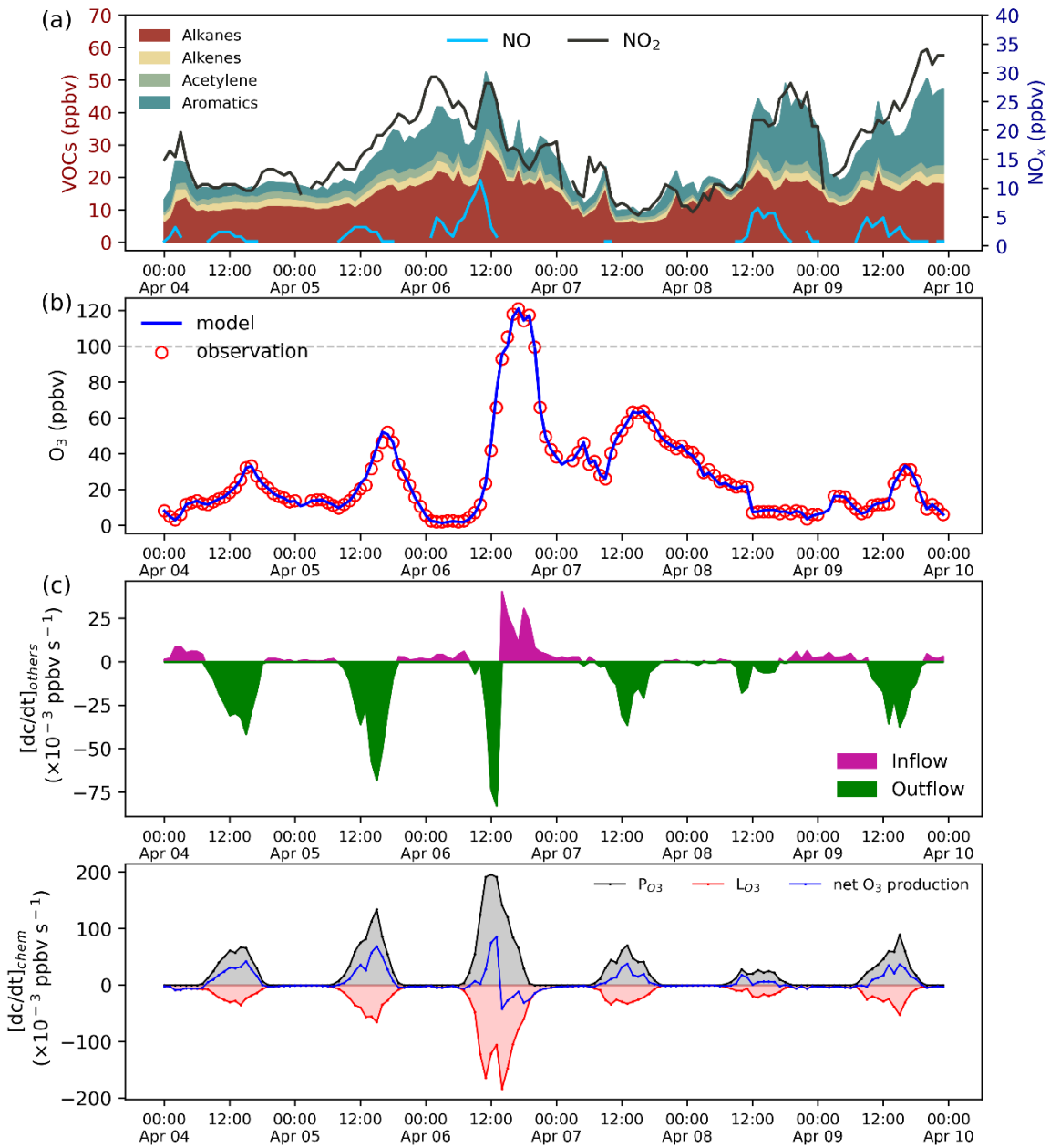



Figure 5. Species mixing ratio and the rate of O₃ change. (a) VOCs and NOx mixing ratios. (b) Model and observation O₃ mixing ratios. (c) The effect of the physical process on the O₃ mixing ratios calculated by the adaptive dynamic optimization module. (d) The rate of O₃ chemical production.

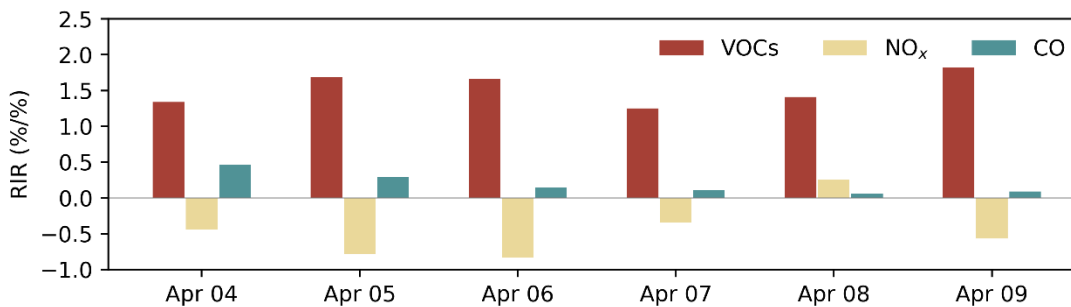



Figure 6. RIR values of O$_3$ precursors, *i.e.,* VOCs, NO$_x$ and CO.

4 Future development

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

— The Ensemble Kalman Filter (EnKF) will be added to dynamically optimize the physical process in future versions.

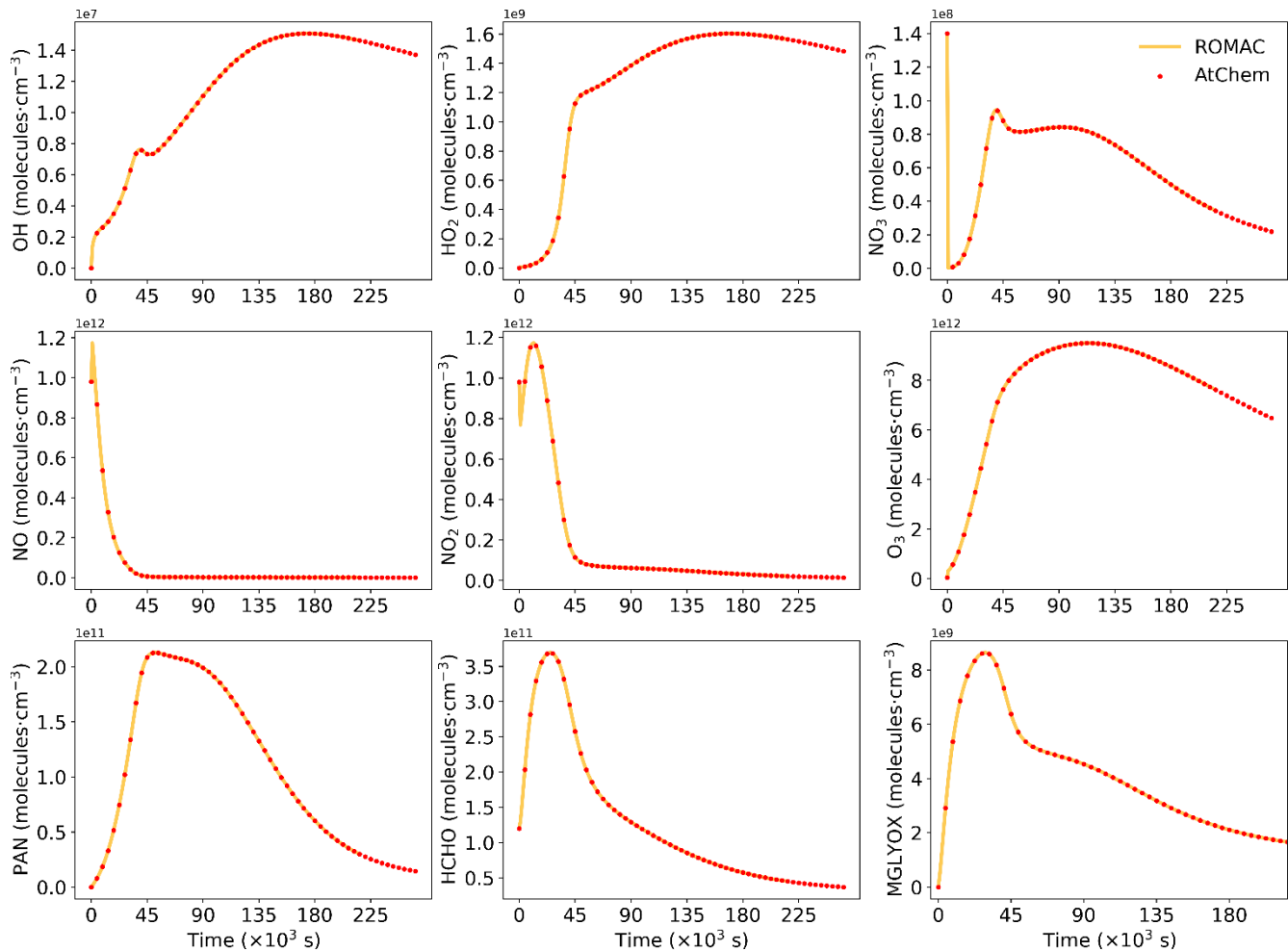
— In future development roadmap, we have plans to introduce a modeling framework version of ROMAC known as "ROMAC-plug-in". This ROMAC-plug-in will support calls from Python or Fortran, ensuring compatibility and flexibility for users. Importantly, the efficient design of ROMAC will be maintained, allowing for optimized performance. The kernel of ROMAC-plug-in will be specifically engineered to provide users with flexibility to effortlessly construct their own models or integrate ROMAC with existing frameworks, such as CTMs.

Appendix A:

Table A1. Initial species concentration used for model comparisons (unit: molecules cm^{-3})

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		

425 **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	*Nonconvergence*	*Nonconvergence*	*Nonconvergence*

430

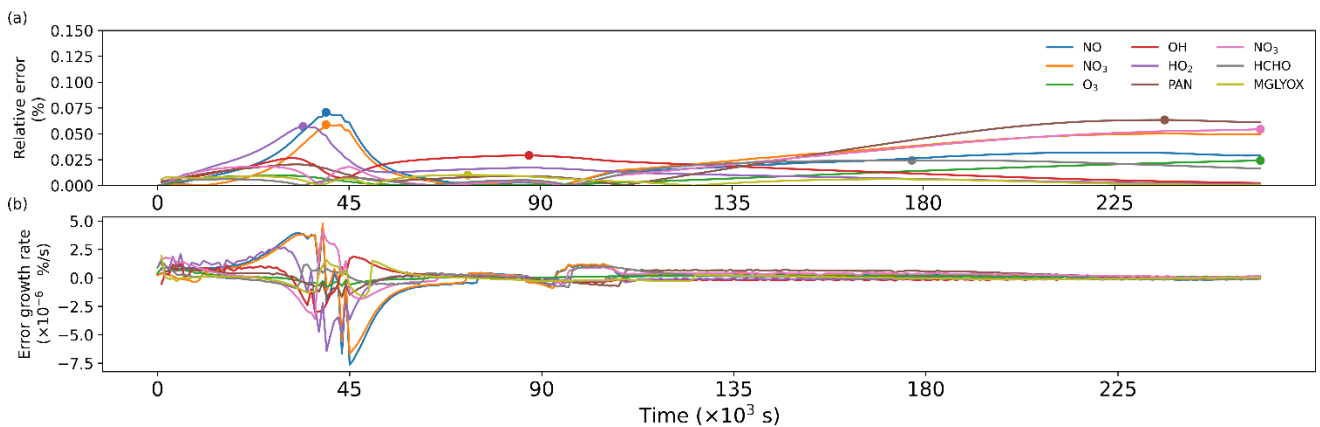
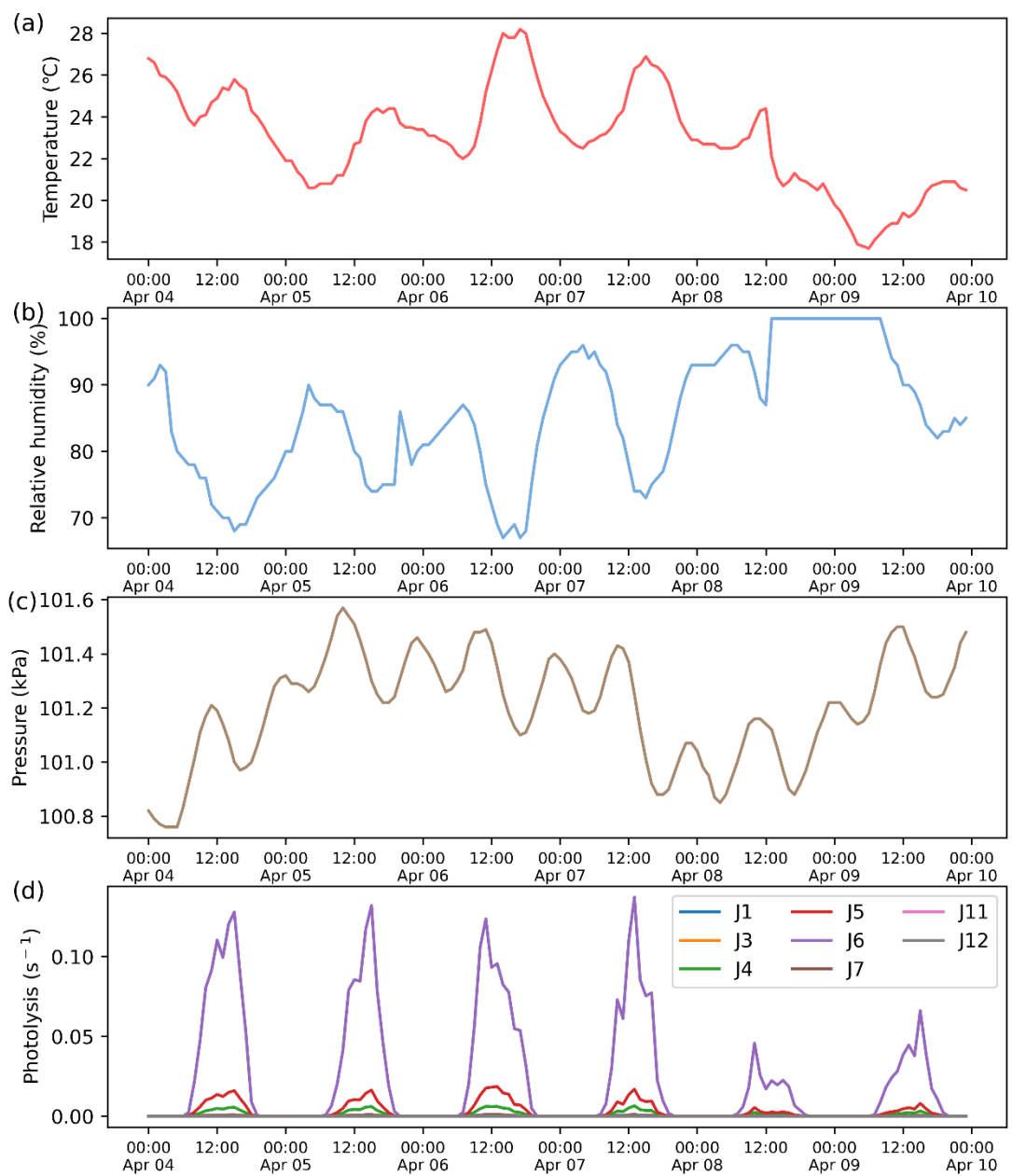



Figure A3. Meteorological data input to the model. (a) temperature. (b) relative humidity. (c) atmospheric pressure. (d) photolysis rate

445 **Appendix B:**

Table B1 Nomenclature

Abbreviations	Explanation
ODEs	Ordinary Differential Equations
VSVOR	The variable-step and variable-order solver
$atol$	absolute tolerance
$rtol$	relative tolerance
r	The reactant in a chemical reaction
p	The product in a chemical reaction
α,β	Stoichiometric number
C_t	Concentration of species at time t
$f_i(C_{i,t},t)$	Rate of change of species i at time t
$P_{i,t}$	Product rate of species i at time t
$L_{i,t}$	Loss rate of species i at time t
$l_{i,t,R}$	The part of the chemical reaction rate that is not directly related to the concentration of species i in reaction R at time t
Δt	Integration time step size
$g_1(C_{t+1})$	The objective function when Newton's method solves the implicit Euler method
$g_2(C_{t+1})$	The objective function when Newton's method solves the implicit trapezoidal method
C_{t+1}^k	Species concentration at iteration k of Newton's method
$\nabla g_1(C_{t+1})$	The Jacobian matrix of $g_1(C_{t+1})$
$\nabla g_2(C_{t+1})$	The Jacobian matrix of $g_2(C_{t+1})$
$\nabla g^{-1}(C_{t+1})$	The inverse of the Jacobian matrix
Δt_0	Integration time step size equal to 2.22×10^{-16} s
Δt_1	Minimum specie atmospheric lifetime in chemical mechanisms
Δt_2	The maximum time step size necessary to achieve diagonal dominance of the Jacobian matrix.
Δt_{init}	Initial integration time step size
Δt_{max}	The maximum integration time step to ensure the result does not exceed the preset tolerance
Δt_{opt}	Optimal integration step size
$RERR$	Relative error calculated by doubled-step method
LTE	Local truncation error
$atol$	Absolute tolerance
$rtol$	Relative tolerance
Rn	Lagrangian remainder in the Taylor expansion
ξ	Real number in the Lagrangian remainder in the Taylor expansion
$RMSE$	Root Mean Square Error

450

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.

455 **Data availability**

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;

460 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.

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

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

Competing interests.

The authors have declared no competing financial interest.

470

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).

475

480   *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.: CHEMSODE: a stiff ODE solver for the equations of chemical kinetics, Comput. Phys. Commun, 97, 304-314,

1996a.

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

490   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

495   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.

500   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.,

505   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

510   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.

Emmons, L. K., Walters, S., Hess, P. G., and Lamarque, J.-F.: Description and evaluation of the Model for Ozone and Related chemical Tracers, version 4 (MOZART-4), Geosci. Model Dev.,, 3, 43-67, 2010.

Esentürk, E., Abraham, N. L., Archer-Nicholls, S., Mitsakou, C., Griffiths, P., Archibald, A., and Pyle, J.: Quasi-Newton
515   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.

520 Hertel, O., Berkowicz, R., and Christensen, J.: Test of two numerical schemes for use in atmospher, Atmospheric Environment, 1993.

Huang, L. and Topping, D.: JlBox 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
525   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 SO2and NH3on the formation of 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,
530   10.1016/s1352-2310(03)00567-3, 2003.

Li, X. B., Fan, G., Lou, S., Yuan, B., Wang, X., and Shao, M.: Transport and boundary layer interaction contribution to extremely high surface ozone levels in eastern China, Environ Pollut, 268, 115804, 10.1016/j.envpol.2020.115804, 2021.

Ling, Z. H., Zhao, J., Fan, S. J., and Wang, X. M.: Sources of formaldehyde and their contributions to photochemical O(3) formation at an urban site in the Pearl River Delta, southern China, Chemosphere, 168, 1293-1301,
535   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, 10.5194/acp-22-2173-2022, 2022.

540   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 HO2 concentrations and their budgets during photooxidation of 2-methyl-3-butene-2-ol (MBO) in the atmospheric

545   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

550   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.

Sander, R., Baumgaertner, A., Cabrera-Perez, D., Frank, F., Gromov, S., Groß, J.-U., Harder, H., Huijnen, V., Jöckel, P.,

555   Karydis, V. A., Niemeyer, K. E., Pozzer, A., Riede, H., Schultz, M. G., Taraborrelli, D., and Tauer, S.: The community atmospheric chemistry box model CAABA/MECCA-4.0, Geoscientific Model Development, 12, 1365-1385, 10.5194/gmd-12-1365-2019, 2019.

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,

560   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.

Sommariva, R., Cox, S., Martin, C., Borońska, K., Young, J., Jimack, P. K., Pilling, M. J., Matthaios, 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

565   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.

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, 2023a.

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.

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., Liu, T., Gong, D., Wang, H., Guo, H., Liao, M., Deng, S., Cai, H., and Wang, B.: Anthropogenic Pollutants Induce Changes in Peroxyacetyl Nitrate Formation Intensity and Pathways in a Mountainous Background Atmosphere in Southern China, Environ Sci Technol, 57, 6253-6262, 10.1021/acs.est.2c02845, 2023b.

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.: Long-term O3–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, 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, 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.