



## PumpKin: A tool to find principal pathways in plasma chemical models<sup>☆</sup>



A.H. Markosyan<sup>a,\*</sup>, A. Luque<sup>b</sup>, F.J. Gordillo-Vázquez<sup>b</sup>, U. Ebert<sup>a,c</sup>

<sup>a</sup> *Centrum Wiskunde & Informatica (CWI), P.O. Box 94079, 1090 GB Amsterdam, The Netherlands*

<sup>b</sup> *IAA-CSIC, P.O. Box 3004, 18080 Granada, Spain*

<sup>c</sup> *Department of Applied Physics, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands*

### ARTICLE INFO

#### Article history:

Received 15 October 2013

Received in revised form

14 March 2014

Accepted 23 May 2014

Available online 2 June 2014

#### Keywords:

PumpKin

Pathway reduction

Chemical pathway analysis

Plasma chemistry

Pathway reduction

Plasma combustion

### ABSTRACT

PumpKin is a software package to find all principal pathways, i.e. the dominant reaction sequences, in chemical reaction systems. Although many tools are available to integrate numerically arbitrarily complex chemical reaction systems, few tools exist in order to analyze the results and interpret them in relatively simple terms. In particular, due to the large disparity in the lifetimes of the interacting components, it is often useful to group reactions into pathways that recycle the fastest species. This allows a researcher to focus on the slow chemical dynamics, eliminating the shortest timescales. Based on the algorithm described by Lehmann (2004), PumpKin automates the process of finding such pathways, allowing the user to analyze complex kinetics and to understand the consumption and production of a certain species of interest. We designed PumpKin with an emphasis on plasma chemical systems but it can also be applied to atmospheric modeling and to industrial applications such as plasma medicine and plasma-assisted combustion.

#### Program Summary

*Program title:* PumpKin

*Catalogue identifier:* AETG\_v1\_0

*Program summary URL:* [http://cpc.cs.qub.ac.uk/summaries/AETG\\_v1\\_0.html](http://cpc.cs.qub.ac.uk/summaries/AETG_v1_0.html)

*Program obtainable from:* CPC Program Library, Queen's University, Belfast, N. Ireland

*Licensing provisions:* Standard CPC licence, <http://cpc.cs.qub.ac.uk/licence/licence.html>

*No. of lines in distributed program, including test data, etc.:* 18753

*No. of bytes in distributed program, including test data, etc.:* 7011133

*Distribution format:* tar.gz

*Programming language:* C++.

*Computer:* All computers with a C/C++ compiler.

*Operating system:* All platforms with a C/C++ compiler.

*Classification:* 16.12, 16.14, 19.11, 19.8.

*External routines:* GLPK (GNU linear programming kit)

*Nature of problem:*

Analyzing complex chemical models.

*Solution method:*

Combining reactions into pathways that recycle short-lived species

<sup>☆</sup> This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

\* Corresponding author. Tel.: +31 614183992.

E-mail addresses: [aram.math@gmail.com](mailto:aram.math@gmail.com), [aram.markosyan@cw.nl](mailto:aram.markosyan@cw.nl) (A.H. Markosyan).

*Running time:*

Runtime depends on the input files sizes and user's configuration. For typical use cases we estimate runtime in the order of minutes.

© 2014 Elsevier B.V. All rights reserved.

## 1. Introduction

Complex chemical models are increasingly demanded by many fields of science. Thanks to the availability of cheap computing power, scientists routinely use models composed of hundreds, if not thousands, of interacting chemical species. Many of these models have a direct relevance for our society's most pressing problems; for example the chemical modeling of the atmosphere plays a significant role in the study of global climatic change [1]. Another example is provided by combustion, where a better understanding of the chemical interactions would lead to better and more efficient fuel use [2].

The demand for complex chemical models is also growing within the plasma physics community. This tendency arises from the recent development of plasma applications involving complex chemical processes, as well as a growing interest in the role of atmospheric electricity on the global chemical balance of our atmosphere [3–7].

The early work by Kossyi et al. published in 1992 [8] can be considered as one of the very first attempts to build a complex kinetic scheme (of about 450 reactions) to study the non-equilibrium plasma kinetics during electric discharges in synthetic air ( $N_2$ – $O_2$  mixtures). In the last 20 years, the employed kinetic schemes have gained in complexity due to the new challenges posed to the scientific community.

For instance, complex kinetic schemes were proposed [9,10] to understand the plasma remediation (through DBD discharges) of  $SO_2$  and  $N_xO_y$  (mainly NO and  $NO_2$ ) resulting from combustion of coal fired electrical power plants and mobile emission sources such as diesel trucks and cars. In addition, the identification of key pathways in the non-equilibrium plasma kinetics of hydrogen–air and hydrocarbon–air mixtures is of great importance for the advance of plasma-assisted combustion (PAC) where hundreds of species and thousands of reactions are usually implemented [11,12].

More recently (since 2005), cold atmospheric pressure plasma jets (APPJs) have been shown as promising tools for biomedical applications [13]. Many of the APPJ plasma species (mixture of air and a noble gas) enable the plasma to disinfect and, consequently, it has become important to study the chemical processes of the plasma jets in order to understand the plasma-induced effects on biological samples. Different models incorporating kinetic schemes of very high complexity have been developed [14–16] including one with almost 2000 reactions [17]; however, none of these works use a systematic tool to trace and analyze the relative importance of the many kinetic pathways considered.

Finally, present studies about the possible electrical activity in the atmospheres of exoplanets are also worth mentioning. In this regard, a recent work by Helling et al. [18] shows first results of the dependence of the electric breakdown conditions in giant gas exoplanets on the local gas-phase chemistry. These preliminary results suggest that different intra-cloud discharges (local corona discharges and large-scale sparks near and above the cloud tops) dominate at several heights inside the mineral clouds of exoplanets. A tool like PumpKin would clearly help in the analysis of the key gas-phase kinetic pathways controlling the atmospheric

chemistry driven by the possible electrical activity in extrasolar planets.

In most cases, when we use a chemical model, we are concerned with the production or consumption of some long-lived species. However, short-lived species very often mediate the chemical interactions between the species of interest so we must also include them in the model. After the model is run, we would like to analyze the result by summarizing only the interactions between long-lived species. This can be done by combining reactions that produce and consume the fast species into a so-called *pathway*. For example, given a fast species  $B$  and the reactions  $A \rightarrow B$ ,  $B \rightarrow C$  these can be combined into the net reaction  $A \rightarrow C$  that recycles the species  $B$ .

This separation between timescales is even more acute in the context of plasma physics and, in particular, in the study of electrical discharges, where also the mechanism driving the chemistry is strongly inhomogeneous in time. This introduces an additional separation of timescales, in this case external to the chemical processes. For example, many plasma applications at atmospheric pressure rely on nanosecond pulsed discharges, where we deal with a strong, repetitively pulsed electric field lasting for some nanoseconds. We may be nevertheless interested in the effects of this electric field on much longer timescales of some milliseconds or more [19].

Many software tools exist to integrate numerically the evolution of arbitrary chemical reaction systems [20–25]. In the plasma context, a tool such as ZDPlaskin [21] incorporates the Boltzmann equation solver BOLSIG+ [26] and thus it is compatible with reaction rates that depend on an evolving electron energy distribution, driven by a time-dependent electric field.

However, few automated tools exist to analyze the output of these codes in terms of the separation between long-lived and short-lived species mentioned above. Recently Lehmann [27] proposed an algorithm to find the most relevant pathways in a given chemical system. With a focus on atmospheric chemistry, this algorithm has been applied successfully to investigate the destruction of ozone in the Earth's atmosphere [28] as well as the formation [29] and stability [30] of  $CO_2$  in the lower Martian atmosphere. Unfortunately, Lehmann [27] did not provide the source code of his implementation of the algorithm to the public access.

We present here the software tool PumpKin (**p**athway **r**eduction **m**ethod for **p**lasma **k**inetic models) that reimplements Lehmann's algorithm with a focus on plasma chemical models. PumpKin is an open-source tool available to all interested researchers, provided that proper reference is made in publications reporting results obtained using this software. The code and its user manual can be downloaded from the web site <http://www.pumpkin-tool.org>. It is also available in the electronic supplement to this paper.

It is worth mentioning that there exists first principle tools (based on quantum chemical calculations) such as the global reaction route mapping (GRRM) using the new ADDF (for one reactant) and AFIR (for two or more reactants) methods for searching elemental reactant pathways recently published by Maeda et al. [31]. Tools such GRRM with the new ADDF and AFIR search methods are of fundamental importance to determine the elemental (chemical)

pathways resulting in some product species as a consequence of the chemical interaction of one (ADDF method), or two or more (AFIR method) reactants. However, our tool (PumpKin) takes the elemental pathways for granted (searched by GRRM through ADDF and AFIR) giving the products of each of the kinetic processes considered (taken from the scientific literature). Thus, PumpKin focuses on the net (or global) reaction pathways producing selected species, as illustrated in Section 4 (examples) for NO and  $O_3^-$ , rather than on the search of possible elemental pathways resulting from the interaction of given reactants which is the principal goal of first principle tools such as GRRM.

The rest of this paper is organized as follows. Section 2 describes the general algorithm of PumpKin. The tool is described in Section 3 where we discuss input and output, as well as technical requirements to run PumpKin. Several examples and use cases are illustrated in Section 4. Our conclusions are detailed in Section 5.

## 2. PumpKin algorithm

In this section we briefly describe the algorithm underlying the PumpKin tool. For a more detailed description and further discussions about the algorithm we refer our users to Lehmann's original paper [27] and to the Users Manual of PumpKin.

Given a model consisting of a set of reactions between several chemical species and the average rates of these reactions, obtained by solving of the model, PumpKin calculates a set of *pathways* and their effective rates. A pathway is a linear combination of reactions, each of them weighted by a natural number. PumpKin builds pathways that have a zero net production of fast species, defined as those with a chemical lifetime much shorter than the time of interest of the model.

To construct a list of pathways, PumpKin proceeds as follows. It starts from a set of pathways consisting of one reaction each, including all the reactions in the model. The list of pathways is then updated in steps, each step treating a fast species (named *branching point* in [27]). At a given point, the next branching point to treat is selected as the species with the shortest lifetime relative to the present list of pathways; then the pathways producing the chosen species are combined with pathways that consume it. The resulting pathways recycle the branching point species, in the sense that they have a zero net production of it. Then, in order to avoid a too large set of pathways, pathways with a low rate are eliminated. Finally, some pathways are decomposed into more elementary ones. This process is repeated for all fast species, defined as those with a lifetime shorter than a user-specified  $\tau$ .

## 3. Code description and general requirements

The PumpKin tool is written in the programming language C++. Originally it has been developed and tested on Mac OS X 10.8.5 using Xcode (versions 4.6.3 and 5.0) with LLVM compilers 4.2 and 5.0. Later it has been successfully compiled with GCC (versions above 4.2) on Mac OS X and Linux. Windows users should have installed the Cygwin tool ([www.cygwin.com](http://www.cygwin.com)), which provides a Linux look and feel environment for Windows. Here are the general requirements:

- To build PumpKin, the GNU version of make (GNUmake) must be installed. The PumpKin makefile requires GNU make version 3.77 or later. GNU software can be downloaded from many places, including [ftp://ftp.gnu.org/gnu/make](http://ftp.gnu.org/gnu/make).
- A C++ compiler is required. PumpKin makes heavy use of the ISO/IEC 14882 C++ Standard. Some compilers are not fully compliant with this specification, although most are. PumpKin

**Table 1**  
Species considered in the model.

N, N <sub>2</sub> , O, O <sub>2</sub> , O <sub>3</sub>
NO, NO <sub>2</sub> , NO <sub>3</sub>
N <sub>2</sub> O, N <sub>2</sub> O <sub>5</sub>
N <sup>+</sup> , N <sub>2</sub> <sup>+</sup> , N <sub>3</sub> <sup>+</sup> , N <sub>4</sub> <sup>+</sup>
O <sup>+</sup> , O <sub>2</sub> <sup>+</sup> , O <sub>4</sub> <sup>+</sup>
NO <sup>+</sup> , N <sub>2</sub> O <sup>+</sup> , NO <sub>2</sub> <sup>+</sup> , O <sub>2</sub> <sup>+</sup> N <sub>2</sub>
N <sub>2</sub> (A <sup>3</sup> Σ <sub>g</sub> <sup>+</sup> , B <sup>3</sup> Π <sub>g</sub> , C <sup>3</sup> Π <sub>u</sub> , a <sup>1</sup> Σ <sub>g</sub> <sup>-</sup> )
N(2D, 2P), O(1D, 1S)
O <sub>2</sub> (a <sup>1</sup> Δ <sub>g</sub> , b <sup>1</sup> Σ <sub>g</sub> <sup>+</sup> , 4.5 eV)
O <sub>2</sub> (X <sup>3</sup> , v = 1–4), N <sub>2</sub> (X <sup>1</sup> , v = 1–8)
e, O <sup>-</sup> , O <sub>2</sub> <sup>-</sup> , O <sub>3</sub> <sup>-</sup> , O <sub>4</sub> <sup>-</sup>
NO <sup>-</sup> , NO <sub>2</sub> <sup>-</sup> , NO <sub>3</sub> <sup>-</sup> , N <sub>2</sub> <sup>-</sup> O

has been compiled and tested with

- GNU g++ 3.32 or higher.
- LLVM 3.2 or higher.
- GLPK (GNU linear programming kit) must be installed [32]. This provides PumpKin a solution of large-scale linear programming (LP) problems. GLPK can be downloaded from website <http://www.gnu.org/software/glpk/>. We have tested PumpKin with GLPK recent version 4.32.

The recommended system requirements depend on the choice of parameters discussed in Section 3.1. As a reference, in a MacBook Pro 15-inch (Mid 2010) with a CPU Intel Core i5 at 2.4 GHz, 4 GB (1067 MHz DDR3) of RAM memory and the operating system Mac OS X 10.8.5, it can analyze a chemical system consisting of 650 reactions and 53 species from Table 1, with the largest possible  $\tau_{lifetime}$  and  $f_{min} = 1000$  (see Section 3.1), in about 30 s. When the input files are large, PumpKin will require more time to load them into the computer memory.

### 3.1. Input and output

To determine the chemical pathways, PumpKin requires from the user the stoichiometric matrix and kinetic data for the full chemical reaction system, namely:

- chemical reactions  $R_j$ ,  $j = 1, \dots, n_R$ , involving between species  $S_i$ ,  $i = 1, \dots, n_S$ , where  $n_R$  and  $n_S$  are the number of chemical reactions and species, respectively,
- stoichiometric coefficients  $s_{ij}$ , which represent the number of molecules of species  $S_i$  produced (or negative number of molecules of  $S_i$  consumed) by reaction  $R_j$ ,
- a time evolution of concentrations  $c_i(t_l)$  and reactions rate  $r_j(t_l)$ , where  $l = 1, \dots, n_T$  and  $t_0 = t_1 \leq \dots \leq t_{n_T} = t_{end}$ .

The code is independent of the units chosen by the user. Conventionally,  $c_i(t_l)$  is specified in units of [mol cm<sup>-3</sup>] and  $r_j$  in units of [mol cm<sup>-3</sup> s<sup>-1</sup>]. A user should also provide an input file similar to the following table

```
interest = 1
t_init = 0.0
t_end = 1.0e-3
max_bp = 0
tau_lifetime = 0.9e-5
max_path = 1500
f_min = 0,
```

(\*)

where:

- *interest*—an index of the species of interest  $S_{interest}$ , if the user is interested in the production and/or consumption of

$S_{\text{interest}}$ , otherwise the user should specify `interest` as a non-positive number,

- `[t_init, t_init]`—a time interval  $[t_{\text{init}}, t_{\text{init}}] \subseteq [0, T]$  where PumpKin will perform the analysis,
- `max_bp`—if positive, the maximum number of branching points considered, otherwise it is disregarded,
- `tau_lifetime`—if positive, a lifetime threshold in units of [s], otherwise it is disregarded,
- `max_path`—if positive, the maximum number of pathways considered per branching point treatment, i.e. the only the first `max_path` pathways with higher rate will be considered, otherwise is neglected,
- `f_min`—if positive, pathway rate threshold in the unit of  $[\text{mol cm}^{-3} \text{s}^{-1}]$ , i.e. pathways with a rate smaller than `f_min` will be deleted, otherwise is neglected.

The order of parameters in the input file should be exactly like in the (\*). The names of parameters are not important.

Currently, PumpKin by default can read an output of ZDPlasKin. The PumpKin package is provided with an example of input files. The output of PumpKin with these input files is discussed in Section 4.

Depending on whether the user has specified the `interest` parameter as a positive integer (the index of the species of interest) or as a non-positive number (the user does not have any species of interest), one of the following results will be printed:

`interest > 0`: PumpKin will output all the pathways (and their rates) producing or consuming the species of interest  $S_{\text{interest}}$ , as well as the relative production or consumption compared with the initial concentration of  $S_{\text{interest}}$ . The output will also contain information such as how much  $S_{\text{interest}}$  has been produced or consumed by the pathways that are deleted by PumpKin using parameters `f_min` or `max_path`.

`interest ≤ 0`: Pumpkin will output all the pathways (and their rates) sorted by rate. In some cases, this number can be very large, so we decided to limit it to 100, which of course can be easily changed inside the PumpKin source code. The output will also contain information such as how much of any species has been produced or consumed by the pathways that are deleted by PumpKin (by user) using parameters `f_min` or `max_path`.

## 4. Examples

PumpKin is provided with two sets of input files. Both examples are the outputs of the zero-dimensional plasma kinetic solver ZDPlasKin. We use a zero-dimensional model to describe the dynamics of species under a constant electric field. The following system of ordinary differential equations (ODEs) is used to describe the interaction between the species

$$\frac{d[n_i]}{dt} = S_i, \quad (1)$$

where the source term  $S_i$  is the total production and destruction rate of species  $i$  in various processes. The adapted version of the kinetic file for  $\text{N}_2\text{-O}_2$  mixtures (dry air) from ZDPlasKin [21,33] is used. A complete list of plasma chemical processes in  $\text{N}_2\text{-O}_2$  mixtures is taken mainly from [34]. Transport parameters and rate coefficients for electron–neutral interactions are calculated using the BOLSIG+ solver built-in into ZDPlasKin. As an initial value of the electron density we take  $n_e(0) = 4.0 \cdot 10^{13} \text{ cm}^{-3}$ .

The list of species and reactions was automatically converted into a system of ordinary differential equations (1) and solved numerically using the ZDPlasKin tool. The user can visualize the results of ZDPlasKin using a software QtPlasKin [35].

### 4.1. Species of interest

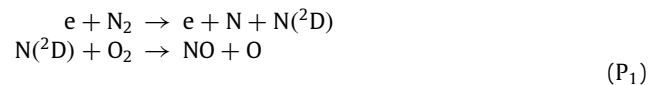
As a first example we assume the background electric field to be a constant 10 kV/cm at standard temperature and pressure (STP). We run our chemical model up to 1 ms with ZDPlasKin. The output is stored in the folder called `Input_10`, where the input parameters specified by the user are in the file `input.txt`, which looks as follows:

```
interest = 40
t_init = 5.0e-11
t_end = 0.9e-3
max_bp = 0
tau_lifetime = 0.9e-5
max_path = 1500
f_min = 0.
```

In this subsection we will modify only the species of interest.

#### 4.1.1. NO

If we run PumpKin for the species NO, PumpKin will output 2 pathways responsible for the production:



net:  $\text{N}_2 + \text{O}_2 \rightarrow \text{N} + \text{O} + \text{NO}$ ,  
and



net:  $\text{O}_3^- + \text{N} \rightarrow \text{O}_2^- + \text{NO}$ .

The pathway ( $\text{P}_1$ ) will produce 52% and pathway ( $\text{P}_2$ ) 41% of the total NO-production, with corresponding rates of  $3 \cdot 10^{13}$  and  $2.4 \cdot 10^{13} \text{ cm}^{-3} \text{ s}^{-1}$ .

The following reaction will be responsible for 72% of the destruction of NO:



and has a rate of  $1.5 \cdot 10^{13} \text{ cm}^{-3} \text{ s}^{-1}$ .

#### 4.1.2. $\text{O}_3^-$

If we run PumpKin and choose  $\text{O}_3^-$  as the species of interest, we obtain the following pathway responsible for 94% of  $\text{O}_3^-$ -destruction



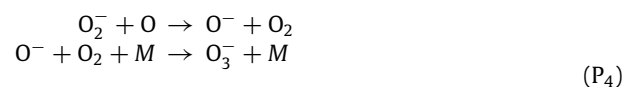
net:  $\text{O}_3^- + \text{O} \rightarrow \text{O}_2^- + \text{O}_2$ ,

with a rate  $3.1 \cdot 10^{16} \text{ cm}^{-3} \text{ s}^{-1}$ .

On the other hand,  $\text{O}_3^-$ -production will be dominated by the following reaction and pathway



and



net:  $\text{O}_2^- + \text{O} + M \rightarrow \text{O}_3^- + M$ ,

where  $M$  is any neutral. The pathway ( $\text{P}_4$ ) will produce 18% of  $\text{O}_3^-$ , with a rate  $1.4 \cdot 10^{16} \text{ cm}^{-3} \text{ s}^{-1}$  and the reaction ( $\text{R}_2$ ) will produce 69% of  $\text{O}_3^-$ , with a rate  $5.3 \cdot 10^{16} \text{ cm}^{-3} \text{ s}^{-1}$ .

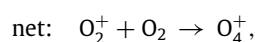
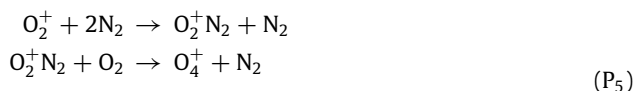


## 4.2. Dominant pathways or reduction

In the second example of input file we consider the electric field to be a constant 20 kV/cm at STP. We run the chemical model up to 2  $\mu$ s. The output is stored in the folder called Input\_20, where the input parameters specified in the file input.txt are as follows:

```
interest = 0
t_init = 5.0e-11
t_end = 2.0e-6
max_bp = 0
tau_lifetime = 2.0e-7
max_path = 1500
f_min = 1.0e+5.
```

The user can benefit from this execution mode in two ways. First by initializing max\_bp by some positive number  $N$  we can eliminate from the system the first  $N$  fastest species. This can significantly reduce the number of species considered in the kinetics. Of course, this depends on the timescale of interest of the user. The second way to use this execution mode is to list the most important pathways sorted by their rate. This also depends on the user's election of where to cut and how many pathways to consider. As a reference, the highest rate of pathway is  $2.4 \cdot 10^{23} \text{ cm}^{-3} \text{ s}^{-1}$  and the 240th pathway has a rate of  $9.9 \cdot 10^{17} \text{ cm}^{-3} \text{ s}^{-1}$ . On the other hand, 26 branching points are already considered, which means that these species are considered as short-lived and recycled. If the user has no interest in these species, he can neglect them. For instance, in this example the first branching point, i.e. the fastest species, is  $\text{O}_2^+ \text{N}_2$  with a lifetime of  $8.5 \cdot 10^{-11} \text{ s}$ . After treating this branching point, PumpKin will generate new pathways recycling it. Among these newly created pathways the following one has one of the highest rates



with a rate of  $3.9 \cdot 10^{19} \text{ cm}^{-3} \text{ s}^{-1}$ . Thus, we see that the pathway ( $\text{P}_5$ ) uses the species  $\text{O}_2^+ \text{N}_2$  only as an intermediate species. PumpKin is pointing here that the chemical model can be reduced by replacing the two original reactions in ( $\text{P}_5$ ) with a single net reaction. The density of  $\text{O}_2^+ \text{N}_2$  does not need to be tracked any longer and therefore the model requires the solution of one equation less.

## 5. Conclusions

The present paper introduces the PumpKin package, a tool to analyze and reduce complex plasma chemical models. The user can analyze the underlying mechanisms of production or/and consumption of the species of interest. The idea is to investigate the relevant chemistry at a given time scale. We have provided two examples of the use of PumpKin as an introduction for potential users that would apply the code to their own models. The present implementation reads only the output format of ZDPlasKin. In principle, though, PumpKin can be extended to read output from other applications.

## Acknowledgments

AHM acknowledges the support from STW-project 10751, part of the Netherlands Organization for Scientific Research (NWO). AHM also acknowledges the support from European Science Foundation (ESF) for short visit grants 5297, 5697 and 5698 within the ESF activity entitled 'Thunderstorm effects on

the atmosphere–ionosphere system'. FJGV and AL acknowledge support by the Spanish Ministry of Economy and Competitiveness (MINECO) under project AYA2011-29936-546-C05-02 and by the Junta de Andalucía, Proyecto de Excelencia FQM-5965.

## References

- [1] N. Butchart, I. Cionni, V. Eyring, T.G. Shepherd, D.W. Waugh, H. Akiyoshi, J. Austin, C. Brühl, M.P. Chipperfield, E. Cordero, M. Dameris, R. Deckert, S. Dhomse, S.M. Frith, R.R. Garcia, A. Gettelman, M.A. Giorgetta, D.E. Kinnison, F. Li, E. Mancini, C. McLandress, S. Pawson, G. Pitari, D.A. Plummer, E. Rozanov, F. Sassi, J.F. Scinocca, K. Shibata, B. Steil, W. Tian, Cchemistry-climate model simulations of twenty-first century stratospheric climate and circulation changes, *J. Clim.* 23 (2010) 5349. <http://dx.doi.org/10.1175/2010JCLI3404.1>.
- [2] S. Turns, *An Introduction to Combustion: Concepts and Applications*, McGraw-Hill, New York, USA, 2012.
- [3] F.J. Gordillo-Vázquez, Air plasma kinetics under the influence of sprites, *J. Phys. D* 41 (23) (2008) 234016. <http://dx.doi.org/10.1088/0022-3727/41/23/234016>.
- [4] D.D. Sentman, H.C. Stenbaek-Nielsen, M.G. McHarg, J.S. Morrill, Correction to "Plasma chemistry of sprite streamers", *J. Geophys. Res. (Atmos.)* 113 (12) (2008) 14399. <http://dx.doi.org/10.1029/2008JD10634>.
- [5] E. Arnone, A. Kero, B.M. Dinelli, C.-F. Enell, N.F. Arnold, E. Papandrea, C.J. Rodger, M. Carlotti, M. Ridolfi, E. Turunen, Seeking sprite-induced signatures in remotely sensed middle atmosphere  $\text{NO}_2$ , *Geophys. Res. Lett.* 35 (2008) 5807. <http://dx.doi.org/10.1029/2007GL031791>.
- [6] F.J. Gordillo-Vázquez, Vibrational kinetics of air plasmas induced by sprites, *J. Geophys. Res.* 115 (2010) A00E25. <http://dx.doi.org/10.1029/2009JA014688>.
- [7] F.C. Parra-Rojas, A. Luque, F.J. Gordillo-Vázquez, Chemical and electrical impact of lightning on the Earth mesosphere: the case of sprite halos, *J. Geophys. Res. (Space Phys.)* 118 (2013) 5190. <http://dx.doi.org/10.1002/jgra.50449>.
- [8] I.A. Kossyi, A.Y. Kostinsky, A.A. Matveyev, V.P. Silakov, Kinetic scheme of the non-equilibrium discharge in nitrogen–oxygen mixtures, *Plasma Sources Sci. Technol.* 1 (1992) 207. <http://dx.doi.org/10.1088/0963-0252/1/3/011>.
- [9] A.C. Gentile, M.J. Kushner, Reaction chemistry and optimization of plasma remediation of  $\text{N}_x\text{O}_y$  from gas streams, *J. Appl. Phys.* 78 (1995) 2074. <http://dx.doi.org/10.1063/1.360185>.
- [10] R. Dorai, M.J. Kushner, Consequences of propene and propane on plasma remediation of  $\text{NO}_x$ , *J. Appl. Phys.* 88 (2000) 3739. <http://dx.doi.org/10.1063/1.1288511>.
- [11] S.M. Starikovskaia, A.Y. Starikovskii, D.V. Zatspein, Hydrogen oxidation in a stoichiometric hydrogen–air mixture in the fast ionization wave, *Combust. Theory Model.* 5 (2001) 97. <http://dx.doi.org/10.1088/1364-7830/5/1/306>.
- [12] A. Starikovskiy, N. Aleksandrov, Plasma-assisted ignition and combustion, *Prog. Energy Combust. Sci.* 39 (1) (2013) 61–110. <http://dx.doi.org/10.1016/j.pecc.2012.05.003>.
- [13] M. Laroussi, X. Lu, Room-temperature atmospheric pressure plasma plume for biomedical applications, *Appl. Phys. Lett.* 87 (11) (2005) 113902. <http://dx.doi.org/10.1063/1.2045549>.
- [14] D.X. Liu, P. Bruggeman, F. Iza, M.Z. Rong, M.G. Kong, Global model of low-temperature atmospheric-pressure He +  $\text{H}_2\text{O}$  plasmas, *Plasma Sources Sci. Technol.* 19 (2) (2010) 025018. <http://dx.doi.org/10.1088/0963-0252/19/2/025018>.
- [15] T. Murakami, K. Niemi, T. Gans, D. O'Connell, W.G. Graham, Chemical kinetics and reactive species in atmospheric pressure helium–oxygen plasmas with humid-air impurities, *Plasma Sources Sci. Technol.* 22 (1) (2013) 015003. <http://dx.doi.org/10.1088/0963-0252/22/1/015003>.
- [16] Q.-Z. Zhang, Y.-X. Liu, W. Jiang, A. Bogaerts, Y.-N. Wang, Heating mechanism in direct current superposed single-frequency and dual-frequency capacitively coupled plasmas, *Plasma Sources Sci. Technol.* 22 (2) (2013) 025014. <http://dx.doi.org/10.1088/0963-0252/22/2/025014>.
- [17] W. Van Gaens, A. Bogaerts, Kinetic modelling for an atmospheric pressure argon plasma jet in humid air, *J. Phys. D* 46 (26) (2013) A265201. <http://dx.doi.org/10.1088/0022-3727/46/27/275201>.
- [18] C. Helling, M. Jardine, C. Stark, D. Diver, Ionization in atmospheres of brown dwarfs and extrasolar planets. III. Breakdown conditions for mineral clouds, *Astrophys. J.* 767 (2013) 136. <http://dx.doi.org/10.1088/0004-637X/767/2/136>.
- [19] S. Nijdam, E. Takahashi, A.H. Markosyan, U. Ebert, Investigation of positive streamers by double-pulse experiments, effects of repetition rate and gas mixture, *Plasma Sources Sci. Technol.* 23 (2) (2014) 025008.
- [20] R.J. Kee, F.M. Rupley, J.A. Miller, *The Chemkin Thermodynamic Data Base*, Tech. Rep. (Mar. 1990).
- [21] S. Pancheshnyi, B. Eismann, G. Hagelaar, L. Pitchford, Computer code ZDPlasKin, University of Toulouse, LAPLACE, CNRS–UPS–INP, Toulouse, France, 2008, <http://www.zdplaskin.laplace.univ-tlse.fr>.
- [22] D. Goodwin, Cantera: object-oriented software for reacting flows, (2002), <http://www.cantera.org>.
- [23] A. Sandu, R. Sander, Technical note: simulating chemical systems in Fortran90 and Matlab with the Kinetic PreProcessor KPP-2.1, *Atmos. Chem. Phys.* 6 (2006) 187.
- [24] N.R. Pinhão, Recent developments on PLASMAKIN—a software package to model the kinetics in gas discharges, *J. Phys. Conf. Ser.* 162 (1) (2009) 012006. <http://dx.doi.org/10.1088/1742-6596/162/1/012006>.

- [25] G.D. Carver, P.D. Brown, O. Wild, The ASAD atmospheric chemistry integration package and chemical reaction database, *Comput. Phys. Comm.* 105 (1997) 197. [http://dx.doi.org/10.1016/S0010-4655\(97\)00056-8](http://dx.doi.org/10.1016/S0010-4655(97)00056-8).
- [26] G.J.M. Hagelaar, L.C. Pitchford, Solving the Boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models, *Plasma Sources Sci. Technol.* 14 (2005) 722. <http://dx.doi.org/10.1088/0963-0252/14/4/011>.
- [27] R. Lehmann, An algorithm for the determination of all significant pathways in chemical reaction systems, *J. Atmos. Chem.* 47 (1) (2004) 45–78. <http://dx.doi.org/10.1023/B:JOCH.0000012284.28801.b1>.
- [28] J.L. Grenfell, R. Lehmann, P. Mieth, U. Langematz, B. Steil, Chemical reaction pathways affecting stratospheric and mesospheric ozone, *J. Geophys. Res. (Atmos.)* 111 (10) (2006) 17311. <http://dx.doi.org/10.1029/2004JD005713>.
- [29] J.W. Stock, C.S. Boxe, R. Lehmann, J.L. Grenfell, A.B.C. Patzer, H. Rauer, Y.L. Yung, Chemical pathway analysis of the Martian atmosphere: CO<sub>2</sub>-formation pathways, *Icarus* 219 (2012) 13–24. <http://dx.doi.org/10.1016/j.icarus.2012.02.010>.
- [30] J.W. Stock, J.L. Grenfell, R. Lehmann, A.B.C. Patzer, H. Rauer, Chemical pathway analysis of the lower Martian atmosphere: the CO<sub>2</sub> stability problem, *Planet. Space Sci.* 68 (2012) 18. <http://dx.doi.org/10.1016/j.pss.2011.03.002>.
- [31] S. Maeda, K. Ohno, K. Morokuma, Systematic exploration of the mechanism of chemical reactions: the global reaction route mapping (GRRM) strategy using the ADDF and AFIR methods, *Phys. Chem. Chem. Phys.* 15 (2013) 3683. <http://dx.doi.org/10.1039/c3cp44063j>.
- [32] A. Makhorin, Gnu linear programming kit, 4.9, <http://www.gnu.org/software/glpk/>.
- [33] A. Flitti, S. Pancheshnyi, Gas heating in fast pulsed discharges in N<sub>2</sub>–O<sub>2</sub> mixtures, *Eur. Phys. J. Appl. Phys.* 45 (2) (2009) 021001. <http://dx.doi.org/10.1051/epjap/2009011>.
- [34] M. Capitelli, C.M. Ferreira, B.F. Gordiets, A.I. Osipov, *Plasma Kinetics in Atmospheric Gases*, Springer Verlag, Berlin, Germany, 2000.
- [35] A. Luque, Computer code QPlaskin, <http://www.trappa.es/content/software>.