How to Generate Species with Positive Concentrations for All Positive Times?

Seyed Mahdi Shavarani[1], János Tóth[*,2,3], Béla Vizvári[4]

[1]*Alliance Manchester Business School, University of Manchester, Manchester, United
Kingdom*
[2]*Department of Analysis, Budapest University of Technology and Economics, Budapest,
Hungary. Fax: +36 1 463 3172; Tel: +36 1 463 2314; E-mail: jtoth@math.bme.hu*
[3]*Chemical Kinetics Laboratory, Institute of Chemistry, Eötvös Loránd University,
Budapest, Hungary*
[4]*Department of Industrial Engineering, Eastern Mediterranean University, Famagusta,
North Cyprus, Turkey*

(Received February 19, 2020)

Abstract

Given a reaction (network) we are looking for minimal sets of species starting
from which all the species will have positive concentrations for all positive times in
the domain of existence of the solution of the induced kinetic differential equation.
We present three algorithms to solve the problem. The first one essentially checks
all the possible subsets of the sets of species. This can obviously work for only a few
dozen species because of combinatorial explosion. The second one is based on an
integer programming reformulation of the problem. The third one walks around the
state space of the problem in a systematic way and produces all the minimal sets of
the advantageous initial species, and works also for large systems. All the algorithms
rely heavily on the concept of Volpert indices, used earlier for the decomposition
of overall reactions (Kovács *et al.,* 2004). Relations to the permanence hypothesis,
possible economic or medical uses of the solution of the problem are analyzed, and
open problems are formulated at the end.

1   Introduction

Suppose we have a mathematical model for a reaction (network), and our goal is that all
the species have positive concentrations for all finite times in their domain of existence.
We would like to achieve this by choosing the initial concentration positive of only a few

species (a minimal set of species). This is to model the real life situation when only the chosen species are put into the reaction vessel at the beginning.

Example 1. Note that being positive even for all finite times does not mean that the stationary concentration of the given species will also be positive, as the simple example X $\longrightarrow$ Y shows, where the stationary concentration of X is zero, still it is positive for all finite times. This example also shows that a species with initially zero concentration can be positive for all positive times: if we start form $x(0) > 0, y(0) = 0$ the concentration $y(t)$ will always be positive after time zero.

Example 2. The example $2\,\mathrm{X} \longrightarrow 3\,\mathrm{X}$ shows that $x(0) > 0$ implies positivity for all positive times in the domain of solution, the interval $[0, 1/x_0[$ which is bounded from above, i.e. **not** the whole set of all nonnegative real numbers.

We shall introduce all our methods on a simple, still not uninteresting example.

Example 3. Let us consider the irreversible Michaelis–Menten reaction:

$$\mathrm{E + S} \rightleftharpoons \mathrm{C} \longrightarrow \mathrm{E + P}$$

In this case it is intuitively obvious that all the species will have positive concentrations if the initial concentrations of the following species are positive:

$$\{\mathrm{C}\}; \{\mathrm{E, S}\}, \{\mathrm{E, C}\}, \{\mathrm{S, C}\}, \{\mathrm{C, P}\}; \{\mathrm{E, S, C}\}, \{\mathrm{E, S, P}\}, \{\mathrm{E, C, P}\}, \{\mathrm{S, C, P}\}; \{\mathrm{E, S, C, P}\}.$$

Of these possibilities the minimal ones are {C} and {E, S}, in the sense that all other sets of species on the list are strict supersets of either set (or even both).

Rephrasing our goal is to create an algorithm producing minimal initial sets of species whose positive initial concentrations are enough to produce all the species in positive concentrations for all finite times. We are looking for effective algorithms using as much chemical information as possible.

The structure of the paper is as follows. In Section 2 we formally introduce the necessary concepts: we give the mathematical definition of reaction networks and their induced kinetic differential equation, the Volpert graph of the reaction, Volpert indexes of species and reaction steps. These definitions can be found in e.g. Ref. [28], but see also our previous paper [13].

Section 3 contains a verbal description—together with further examples—of a refined version of the brute force solution. Section 4 provides arguments for an integer programming formulation. To treat really large problems we need a smart solution taking into consideration the structure of the problem. This is given in Section 5. Relevant parts of the codes for all the methods are presented either as pseudocodes, or as real codes. More details can be found in our notebook written in the `Wolfram language` or our Matlab worksheets available from the authors upon request. Unusual it may be but not earlier than before the end shall we analyze possible applications and relations of the problem with other fields of kinetics; this would not have been easy to follow earlier. Note that our codes heavily use the `Wolfram language` package `ReactionKinetics` downloadable from the site `http://extras.springer.com`.

To end our introduction, we enumerate the notations used in the paper.

Table 1. Notations

Notation	Meaning
$\mathcal{A}$	the set of atoms
$\mathbf{c}$	concentration vector at time t
$c_m(t)$	concentration of species X(m) at time t
C	complex
E	enzyme
k_r	reaction rate coefficient
$\mathcal{M}$	the set of labels of species
$\mathcal{M}_0$	the set of labels of initial species
m	label of species
M	number of species
$\mathbb{N}$	the set of positive integers
P	product
$\mathcal{R}$	the set of labels of reaction steps
r	label of reaction step
R	number of reaction steps
$\mathbb{R}$	the set real numbers
$\mathbb{R}_+$	the set positive real numbers
S	substrate
X	species
X(m)	species
$y(t)$	concentration of species Y at time t
Y	species
$\alpha(m,r)$	reactant stoichiometric coefficient
$\beta(m,r)$	product stoichiometric coefficient
μ	the largest Volpert index of species
ϱ	the largest Volpert index of reaction steps

2   Problem formulation. Volpert indices

Here we shortly introduce the used formalism, the details can be found e.g. in [8] or [28]. Suppose we have a reaction vessel of constant volume, pressure and temperature and suppose it contains the chemical species $X(1), X(2), \ldots X(M)$, where $M \in \mathbb{N}$ is a positive integer, and suppose that among the species the following reaction steps take place:

$$\sum_{m \in \mathcal{M}} \alpha(m, r) X(m) \longrightarrow \sum_{m \in \mathcal{M}} \beta(m, r) X(m) \quad (r \in \mathcal{R}) \tag{2.1}$$
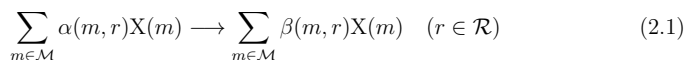
with $\mathcal{M} := \{1, 2, \cdots M\}$ and $\mathcal{R} := \{1, 2, \cdots R\}$, where $R \in \mathbb{N}$; and the elements of the matrices $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are supposed to be nonnegative integers, they are called **stoichiometric coefficients**. The object of our investigation is the mass action type kinetic differential equation of the reaction

$$\dot{c}_m = \sum_{r \in \mathcal{R}} (\beta(m, r) - \alpha(m, r)) k_r \prod_{p \in \mathcal{M}} c_p^{\alpha(p, r)} \quad (m \in \mathcal{M}) \tag{2.2}$$

describing the time evolution of the vector of concentrations $\mathbf{c} = [c_1, c_2, \cdots c_M]^\top$ of the species, where $k_1, k_2, \cdots, k_R \in \mathbb{R}_+$ are the **reaction rate coefficient**s.

There are different graphs representing the reaction [28, Chapter 3] of which we shall use the one actually introduced by Petri in 1939. However, it was A. I. Volpert [30] who started to connect properties of this graph to the properties of the solutions of induced kinetic differential equations. Further applications of this graph on the qualitative behavior of the solutions of (2.2) can be found in [2, 5, 7, 9, 11], or on the decomposition of overall reactions [13]. (One can also meet rediscoveries in the literature: [10, 24].)

Definition 1. The Volpert graph of the reaction (2.1) is a directed bipartite graph with the species and reaction steps as vertices, and with $\alpha(m, r)$ edges going from $X(m)$ to r, and with $\beta(m, r)$ edges going from r to $X(m)$.

Note that this is the graph most often used in papers and textbooks on metabolism. Fig. 1 shows the Volpert graph of the Michaelis–Menten reaction.
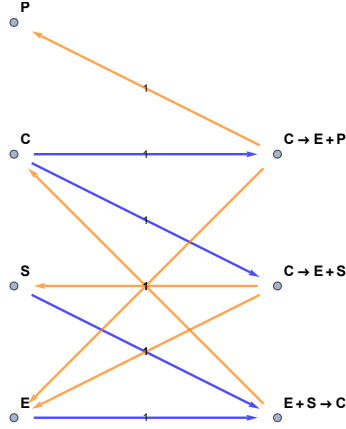
Figure 1. The Volpert graph of the Michaelis–Menten reaction

Next we introduce indexing of the vertices of the Volpert graph.

Definition 2. Let $\mathcal{M}_0 \subset \mathcal{M}$ be a subset of the species index set, called the indices of **initial species**.

1. Elements of $\mathcal{M}_0$ receive zero index. Zero will also be assigned to those reaction steps r for which all the reactant species are available; in this case this means that all the reactant species are in $\mathcal{M}_0$.

2. Suppose that indices up to $k \in \mathbb{N}_0$ has been allotted. Then, a species receives the index $k+1$, if it has no index yet and there is a reaction step producing it. A reaction step receives the index $k + 1$ if it has no index yet and all the species needed for it to take place are available; in this case this means that all the reactant species have an index not more than k.

3. As the total number of vertices is finite, the algorithm ends after a finite number of steps. Species and reaction steps without index at termination will receive an index $+\infty$.

Remark 1. Notice that the value of the indices does not depend on the values of the stoichiometric coefficients, what only counts is their positivity. I.e. if one has two reaction networks with stoichiometric coefficients that are positive for the same indices m, r then the Volpert indices of the species and reaction steps will be the same. Our topics is

quite different from the case when differences in the values of stoichiometric coefficients lead to qualitatively different consequences, e.g. [8, p. 112]. The time evolution of the concentrations also do depend on the values of the stoichiometric coefficients.

Obviously, in the case when all the vertices receive a finite index, then the largest index ϱ a reaction step can have is smaller than R; the largest index μ a species step can have is smaller than M, and either $\varrho = \mu$ or $\varrho + 1 = \mu$. The value of the Volpert index shows a kind of distance of a vertex from the initial species, [28, Theorem 9.1]. When applying the definition, the initial set of species will be those with positive concentration at the beginning.

Let us see examples of indexing in the case of the Michaelis–Menten reaction.

Example 4. First, assume that E and S were initially present, then we get the following indices.

Species	Indexes
S	0
E	0
C	1
P	2

Reaction steps	Indexes
$E + S \longrightarrow C$	0
$C \longrightarrow E + P$	1
$C \longrightarrow E + S$	1

Now a less favorable choice follows: the species initially present is P.

Species	Indexes
P	0
C	$+\infty$
S	$+\infty$
E	$+\infty$

Reaction steps	Indexes
$C \longrightarrow E + P$	$+\infty$
$C \longrightarrow E + S$	$+\infty$
$E + S \longrightarrow C$	$+\infty$

Let us consider a more complicated example.

**Example 5.** We take it from [18], but it can also be found in the Biomodels Database; see also [15].

Species	Indexes
X_{10}	0
X_7	0
X_{11}	1
X_5	1
X_3	1
X_2	1
X_4	1
X_9	2
X_8	2
X_1	2
X_6	3

Reaction steps	Indexes
$X_{10} \longrightarrow X_{11}$	0
$X_{10} \longrightarrow X_2 + X_5$	0
$X_7 \longrightarrow X_3 + X_4$	0
$X_7 \longrightarrow X_2 + X_4$	0
$X_{11} \longrightarrow X_1 + X_5$	1
$X_2 + X_5 \longrightarrow X_{10}$	1
$X_2 + X_5 \longrightarrow X_9$	1
$X_3 + X_5 \longrightarrow X_8$	1
$X_2 + X_4 \longrightarrow X_7$	1
$X_1 + X_5 \longrightarrow X_{11}$	2
$X_9 \longrightarrow X_2 + X_5$	2
$X_8 \longrightarrow X_9$	2
$X_8 \longrightarrow X_3 + X_5$	2
$X_1 \longrightarrow X_4 + X_6$	2
$X_6 \longrightarrow X_2 + X_4$	3
$X_6 \longrightarrow X_1 + X_4$	3

where the meaning of the species are as follows. MAPK is the abbreviation for mitogen-activated kinase in the table.

Notation 1	Notation 2	Meaning
X_1	M	dephosphorylated MAPK
X_2	Mp	MAPK phosphorylated on one residue
X_3	Mpp	MAPK phosphorylated on both residues
X_4	MAPKK	MAPK kinase
X_5	MKP	MAP kinase phosphatase
X_6	M−MAPKK	Complex
X_7	Mp−MAPKK	Complex
X_8	Mpp−MKP	Complex
X_9	Mp−MKP	Complex
X_{10}	Mp−MKP·	Complex
X_{11}	M−MKP	Complex

Now we give another kind of representation of this reaction, this time by the Feinberg–Horn–Jackson graph shown in Fig. 2. This representation can surely be reproduced by the reader without a formal definition; one should only remember that all the **complexes** (formal linear combinations on both sides of the reaction arrows) should only appear once in the figure. Although all our calculations rely on the Volpert graph, the Feinberg–Horn–Jackson graph may be more transparent in many cases.
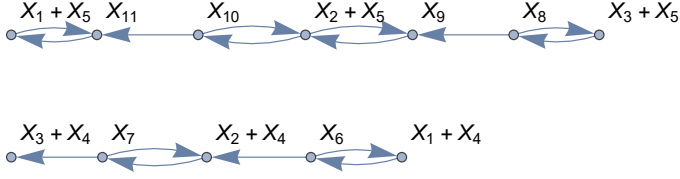
Figure 2. The Feinberg–Horn–Jackson graph of the mitogen-activated protein kinase cascade

In order to formulate our problem we need a theorem by Volpert, see e.g. [28, Theorems 8.13 and 8.14].

Theorem 1. Suppose a set of species $\mathcal{M}_0 \subset \mathcal{M}$ with positive initial concentrations is given.

1. For all species $X(m) \in \mathcal{M}$ with a finite index and for all times t in the domain of existence of the solution of the induced kinetic differential equation $c_m(t) > 0$ holds.

2. If the zero complex is not a reactant complex in the reaction (2.1), then for all species $X(m) \in \mathcal{M}$ with an infinite index and for all times t in the domain of existence of the solution $c_m(t) = 0$.

Cf. the content of the theorem to the behavior of absolute probabilities of a Markovian pure jump process ([16, Remark 2.48]), but note that it is a slightly different problem form the mathematical point of view: it relates an infinite system of linear differential equations while (2.2) is a finite system of nonlinear equations.

If the zero complex is a reactant complex then the function `VolpertIndexing` treats it as if it were a species automatically present in the set of initials, and the first part of the theorem still applies. The result of

$$\texttt{VolpertIndexing[\{0 -> X\}, \{ \}, Verbose -> True]}$$

is as follows.

Species	Indexes
0	0
X	1

Reaction steps	Indexes
$0 \longrightarrow X$	0

showing that the (single) reaction step receives the index 0, because it starts from 0, supposed to be present.

Now our problem can be reformulated as finding such minimal initial set(s) with which the indices of all the species will be finite.

3 Brute force tamed

First, we describe and apply the brute force method, then we analyze its performance bounds.

3.1 Steps of the brute force method

Our first strategy is to form all possible nonempty subsets of $\mathcal{M}$ (the power set except the empty set), and select those with which as initial species all the species have a finite Volpert index. Then select the minimal sets from the above set. Although our calculations rely on the Volpert graph, the Feinberg–Horn–Jackson graph is more transparent in many cases. Selecting the minimal set means that given a set of sets we construct the underlying Sperner family (antichain or clutter) from it. (Sperner family is a set of subsets of a finite set such that none of the element of the family contains another element of the family as a subset [17].) An easy to read Wolfram Language code for this (`MinimalOnes`) can be found in the Appendix, which works as expected.

```
MinimalOnes[{{C}, {E, S}, {E, C}, {S, C}, {C, P}, {{E, S, C}, {E, S, P},
{E, C, P}, {S, C, P},{E, S, C, P}}]
```

gives $\{\{C\},\{E,S\}\}$. Now suppose we are given a reaction network and a set of initial species $\mathcal{M}_0$ and we wonder whether all the species are generated from $\mathcal{M}_0$. This happens if all the Volpert indexes of the species are finite.

```
FiniteVolpertIndexQ[{"E","S"}, "Michaelis-Menten"]
```

gives the expected result: `True`, whereas

```
FiniteVolpertIndexQ[{"E"}, "Michaelis-Menten"]
```

will obviously be `False`.

Before calculating the power set of the species we introduce a few selection principles.

1. If the reaction is not reversible then product species which are not reactant species at the same time, need not be taken into consideration. An example is the species P in the Michelis–Menten reaction. This is a requirement that can be realized in a purely formal, algorithmic way; from now on we shall start with the reactant species only. These are determined by `ReactantSpecies`.

   The result of `ReactantSpecies["Michaelis-Menten"]` will be $\{C,E,S\}$ as expected.

2. The help of the chemist is needed to eliminate **intermediates**, i.e. species which cannot be present initially because of chemical reasons. This set should always be specified by the user, although we could include such species as OH forever, or we want to include C in case of the Michaelis–Menten reaction. As to the second case: it would be desirable to algorithmically find such kind of intermediates, because they are defined by the structure of the reaction and not by the chemical nature of C.

3. A formal, automatic treatment can also be used to throw away **subsets** of species where not all the atoms present in the reaction are contained in the species of the set.

   A working code can again be found in the Appendix. If

   $$\text{EmanuelKnorre} = \{Cl_2 \to 2Cl^*, CH_4 + Cl^* \to {}^*CH_3 + HCl, {}^*CH_3 + Cl_2 \to CH_3Cl + Cl^*\}$$

   then `AtomsPresentQ[EmanuelKnorre, {HCl}]` will give `False`, while

   $$\text{AtomsPresentQ[EmanuelKnorre, \{HCl, C\}]}$$

   provides `True`.

Now one can collect the species which can be present in the initial sets.

$$\text{PowersetBase[reac\_,intermediates\_:\{\}] :=}$$
$$\text{Complement[ReactantSpecies[\{reac\}], intermediates]}$$

The result of `PowersetBase["Michaelis-Menten", {"C"}]` is `{"E","S"}` as expected, because P as a terminal species has been discarded by the program, and the chemist asked us to discard the intermediate C. When forming the subsets of species one applies the check for the presence of all the necessary atoms. The final code has been collected in the Appendix.

Here we give a few results, our notebooks contain a large set of examples.

$$\text{MinimalInitials["Michaelis-Menten"]}$$

gives `{{"C"},{"E", "S"}}`, and

$$\text{MinimalInitials["Michaelis-Menten",\{"C"\}]}$$

gives $\{\{$"E", "S"$\}\}$.

MinimalInitials[EmanuelKnorre], MinimalInitials[EmanuelKnorre,{}],

finally MinimalInitials[EmanuelKnorre,{},Atomic→True] all give $\{\{$CH$_4$,Cl$_2\}\}$, but evaluation of the last command takes (in this case: only slightly) less time.

Finally, a larger example follows.

MinimalInitials[MAPKBIOMD26] provides multiple solutions:

$$\{X_{10}, X_4\}, \{X_{10}, X_6\}, \{X_{10}, X_7\}, \{X_{11}, X_4\}, \{X_{11}, X_6\}, \{X_{11}, X_7\},$$
$$\{X_4, X_8\}, \{X_4, X_9\}, \{X_5, X_6\}, \{X_5, X_7\}, \{X_6, X_8\}, \{X_6, X_9\},$$
$$\{X_7, X_8\}, \{X_7, X_9\}, \{X_1, X_4, X_5\}, \{X_2, X_4, X_5\}, \{X_3, X_4, X_5\}.$$

Further examples can be found in the notebook containing all the calculations.

## 3.2 Performance bounds of the brute force method

Let us start with the hardware. The test problems were solved on a laptop with 16 GB RAM and Intel® Core™ i7-8550U CPU 1.80 GHz 1.99 GHz running a 64-bit Windows 10 operating system.

Two reasons may limit the method: the time needed to do the calculations and the memory needed. It turns out immediately that the second one is the real problem, therefore we dwell on this longer.

First, we measured the time needed to form the power set of sets having different sizes: Fig. 3 It will turn out later that the reason we could not measure a timing for the 26 element set is that there is not enough memory space to store its power set.
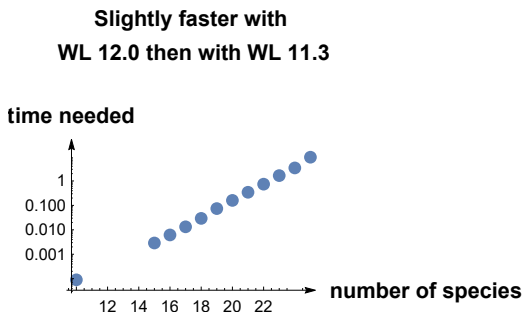
Figure 3. Times needed (from measurements) to form the power sets of sets having different sizes
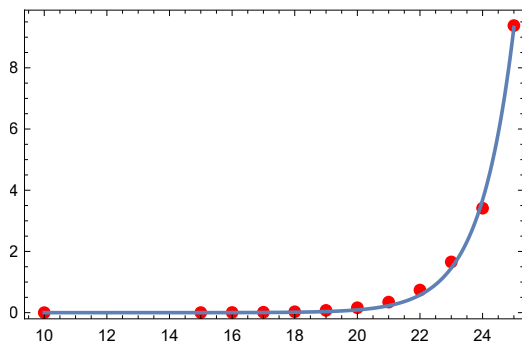
Figure 4. Curve fitted to measured times needed to form the power sets of sets having different sizes

(We have also found that Wolfram Language 12.0 is slightly faster than version 11.3.) Extrapolation from fitting a curve to the data (Fig. 4) shows that with 30 element to form the subsets 16 minutes would be needed, with 40 elements 4 months, and with 50 elements 3500 years, a bit too long time.

The typical value of the variable `MemoryAvailable[]` is 12,747,689,984 on our computer, and `Total[Map[ByteCount, Subsets[Range[25]]]]` is 11,506,401,520, thus, without other ideas reactions with more than 25 species cannot be treated in such a simple way.

The final consequence should be that it is worth constructing other algorithms, as well.

4   Using integer linear programming

First, we have to introduce a series of further notations.

4.1   Notations

Let Reactants(r) denote the species in the reactant complex of reaction step $r \in \mathcal{R}$:

$$\text{Reactants}(r) := \{X(m) \in \mathcal{M}; \alpha(m, r) > 0\},$$

and Products(r) the species in the product complex of reaction step $r \in \mathcal{R}$:

$$\text{Products}(r) := \{X(m) \in \mathcal{M}; \beta(m, r) > 0\}.$$

Furthermore, let Producing(m) be the set of those reaction steps which produce the species $m \in \mathcal{M}$:

$$\text{Producing}(m) := \{r \in \mathcal{R}; \beta(m,r) > 0.\}$$

We have a few binary variables (variables which can only have the value 0 or 1): $y(m,\ell) = 1$ if and only if species X(m) has a Volpert index ℓ, otherwise it is zero; $z(r,\ell) = 1$ if and only if reaction step $r \in \mathcal{R}$ has a Volpert index $\ell - 1$.

4.2 Formulation of the problem

Here we sketch how our problem can be attacked using the methods of integer programming problems. After the presentation of the method we analyse what is missing and what is different in this formulation compared to the original problem.

Find the minimum of the number of initial species

$$\mathcal{O}(\mathbf{y}, \mathbf{z}) := \sum_{m=1}^{M} y(m, 0)$$

in the variables $\mathbf{y}, \mathbf{z}$ under the conditions (4.1)–(4.9) below. The meaning of the value of the objective function $\mathcal{O}$ is the total number of different initial species.

$$\forall m \in \mathcal{M} \forall \ell \in \mathcal{R} : y(m, \ell) \text{ is 0 or 1;} \tag{4.1}$$

$$\forall m \in \mathcal{M} \forall \ell \in \mathcal{R} : z(r, \ell) \text{ is 0 or 1.} \tag{4.2}$$

Eqs. (4.1) and (4.2) express the requirement that the variables are binary variables.

$$\forall m \in \mathcal{M} : \sum_{\ell=0}^{R-1} y(m, \ell) = 1; \tag{4.3}$$

Condition (4.3) means that all the species are produced with some finite Volpert index.

$$\forall r \in \mathcal{R} : \sum_{\ell=0}^{R-1} z(r, \ell) = 1; \tag{4.4}$$

Eq. (4.4) means that all the reaction steps take place (i.e. they have a finite Volpert index). Note that if the Volpert index of all the species is finite then no reaction step can have an infinite Volpert index.

From now on the absolute value sign is used to denote the number of elements of a finite set.

$$\forall r \in \mathcal{R} \forall k \in \mathcal{R} : \sum_{m \in \text{Reactants}(r)} \sum_{\ell=0}^{k} y(m, \ell) - \sum_{\ell=1}^{k} z(r, \ell) \leq |\text{Reactants}(r)| - 1; \tag{4.5}$$

If all the reactants are present, then the reaction step can occur.

$$\forall r \in \mathcal{R} \forall k \in \mathcal{R} : \sum_{m \in \text{Reactants}(r)} \sum_{\ell=0}^{k-1} y(m, \ell) - |\text{Reactants}(r)| \sum_{\ell=1}^{k} z(r, \ell) \geq 0; \qquad (4.6)$$

If any of the reactant is absent, then the reaction step cannot occur.

$$\forall r \in \mathcal{R} \forall k \in \mathcal{R} \forall m \in \text{Products}(r) : -\sum_{\ell=0}^{k} y(m, \ell) + z(r, k) \leq 0; \qquad (4.7)$$

If the reaction step r has the Volpert index k, then at least one of the species has a Volpert index not larger than k.

$$\forall r \in \mathcal{R} : \sum_{\ell=1}^{R} z(r, \ell) \geq 1; \qquad (4.8)$$

The Volpert index of all the reaction steps is finite.

$$\forall m \in \mathcal{M} \forall k \in \mathcal{R} : \sum_{\ell=1}^{k} y(m, \ell) \leq \sum_{r \in \text{Producing}(m)} \sum_{\ell=1}^{k} z(r, \ell). \qquad (4.9)$$

The Volpert indices of the product species of a reaction step are not greater than the Volpert index of the reaction step plus 1.

First, note that the minimum of the number of initial species does not define the same initial set that we called minimal above. Second, further conditions can be formulated to eliminate end products and intermediates, etc. Instead of refining this approach we provide another one that works for large data sets and fits most naturally to the structure of the problem.

5 The smart solution

First, we describe lexicographic and reverse lexicographic orderings, then we show how to apply them to the solution of our problem.

5.1 The lexicographic and reverse lexicographic orderings as tools of enumeration

To obtain the solution of many different problems a large number of objects must be enumerated. The enumeration must be fast and exact. An enumeration is fast if no object is considered more than once and it is exact if all relevant objects are considered. Brute force is not fast as it considers many irrelevant objects. Thus, the enumeration must

consider every investigated object exactly once. Notice that this kind of enumeration gives a complete order of the objects. In a complete order, any two different objects are comparable, i.e. one of them precedes the other one.

There are two types of constituents in a chemical reaction network, species and reaction steps. (We may—and shall—also take into consideration that species are built up from atoms.) The problem of determining the minimal sets of initial species in such a way that starting from these sets, the reaction steps produce all species, needs the enumeration only one of the two subsets, i.e. either the subsets of species, or the subsets of reaction steps. We shall be working with the first.

These subsets, i.e. the subsets of species are the objects to be enumerated. Assuming that there are $M := |\mathcal{M}|$ species, the total number of all the subsets is 2^M, which can be very huge. Taming brute force meant to exclude some species or some subsets of species based on preliminary considerations, not an essential speed-up. Thus, an effective enumeration method must handle the majority of cases in an implicit way. It means that the individual investigation of the implicitly enumerated cases is not necessary. For example, if a subset T satisfies all the requirements, then the investigation of any subset U such that $T \subset U$ is superfluous.

The subsets must be represented in a way which is suitable for calculation. The **characteristic vector** of the subsets is one possible tool. The characteristic vector u of a subset $\mathcal{M}_0 \subset \mathcal{M}$ is an M-dimensional vector such that

$$u_m := \left\{ \begin{array}{ll} 1, & \text{if } m \in \mathcal{M}_0; \\ 0, & \text{if } m \notin \mathcal{M}_0. \end{array} \right.$$

Such vectors can be ordered by the lexicographic order of vectors. A vector such that every component is either 0, or 1, is called **binary** or **0-1** or **Boolean vector**. The number of binary vectors of fixed dimension is finite. Let us introduce a lexicographically increasing order of such vectors.

5.1.1 Lexicographic ordering.

Definition 3. Let $u, w \in \{0,1\}^M$ be two M-dimensional vectors. Vector u is **lexicographically smaller** than vector w if an index $p : 1 < p \leq M$ exists such that

1. $u_m = w_m$ for $m = 1, \ldots, p-1$, and

2. $u_p < w_p$.

This relation is denoted by $u \prec w$.

Therefore, the lexicographically the smallest vector is $(0, 0, \ldots, 0, 0)$, and the second smallest vector is $(0, 0, \ldots, 0, 1)$. Assume that u is a non-zero binary vector, different from the one only having 1 as coordinates. Let p $(1 \leq p < M)$ be the last index such that $u_p = 0, u_{p+1} = 1$. In the lexicographical order, the **next** vector is v with $u_m = v_m$ $(m = 1, \ldots, p - 1)$, $v_p = 1, v_{p+1} = \ldots v_M = 0$. Consequently, the vector such that all coordinates are 1, is the lexicographically largest binary vector, and no vector exists thereafter.

Note that lexicographic ordering orders the binary vectors in the same way as if they were ordered according to the values of the number composed from the components as digits in the binary system.

5.1.2 Reverse lexicographic ordering.

Another ordering, called **reverse lexicographic ordering** is obtained if the coordinates are used in the opposite order.

Definition 4. Let $u, w \in \{0, 1\}^M$ be two M-dimensional vectors. Vector u is **smaller with respect to the reverse lexicographic ordering** than vector w if an index p : $1 < p \leq M$ exists such that

1. $u_m = w_m$ for $m = p + 1, \ldots, M$, and
2. $u_p < w_p$.

If the reverse lexicographic order of the coordinates is used, then the rule is as follows: the first 0 from left is changed to 1 and all coordinates before it must be zero. Assume that the current vector is u with $u_1 = \cdots = u_p = 1, u_{p+1} = 0$. Subsequently, the next vector v is such that $v_1 = \cdots = v_p = 0, v_{p+1} = 1$ and for $m = p + 2, ..., M : v_m = u_m$. Specifically, $(0, 0, ..., 0, 0)$ is the smallest, and $(1, 1, ..., 1, 1)$ is the largest vector also in the case if reverse lexicographic order is applied. However, the second and third smallest vectors are $(1, 0, ..., 0, 0)$, and $(0, 1, ..., 0, 0)$, respectively. In what follows, this **reverse lexicographic order** will also be used. See Table 2. Using again the Michaelis–Menten example we find that Case 5 precedes Case 6, meaning that the potential initial set $\{C\}$ precedes the initial set $\{E, C\}$.

Note that beyond the two orderings above any other order of the coordinates gives another lexicographic order.

Case	Vector	Case	Vector
1	(0,0,0,0)	9	(0,0,0,1)
2	(1,0,0,0)	10	(1,0,0,1)
3	(0,1,0,0)	11	(0,1,0,1)
4	(1,1,0,0)	12	(1,1,0,1)
5	(0,0,1,0)	13	(0,0,1,1)
6	(1,0,1,0)	14	(1,0,1,1)
7	(0,1,1,0)	15	(0,1,1,1)
8	(1,1,1,0)	16	(1,1,1,1)

Table 2. The enumeration of the 16 binary vectors of the 4-dimensional space if the reverse lexicographic order is used.

5.2 The application of reverse lexicographic ordering to the enumeration of minimal sets of species

The logic of the enumeration in the investigation of a chemical reaction is as follows. Two actions must be done with every vector u.

The first one is the checking if all materials have a finite index in the Volpert graph given that u is the characteristic vector of initial species. One possible answer to this question is given by the procedure **complete** in the pseudocode of Fig. 6, or the above used Boolean function `FiniteVolpertIndexQ` defined in the Appendix 7. Its value is true if the condition is satisfied, otherwise the value is false.

The second one is that the vector immediately following u is to be generated. It is done by the function **next**(u). The minimal solutions are collected in set U. The algorithm puts a vector u into this set, if the vector represents a subset as its characteristic vector which is currently minimal. If there are vectors in U which represent subsets larger than the subset represented by u, then they must be deleted from U.

The rule how to obtain the lexicographically next vector is as follows: Find the last 0 in the vector, i.e. the 0 component with the highest index. Thus, the end of the vector is $(\ldots, 0, 1, \ldots, 1)$. Change the zero to 1 and behind it all components become 0. The code for **next**( ) in the case of reverse lexicographic ordering is similar.

With the code in Fig. 6 the problem has been solved, unless one wants to take into consideration the atomic structure of species to reduce the time needed to the calculations.

```
1 for m := M step −1 to 1 do
2    if u_m = 0
3    then
4    begin
5        u_m := 1
6        for p := m + 1 to M do u_p := 0
7    end
```

Figure 5. This is the code for the function **next** with lexicographic ordering

```
1 begin
2    u := (0, 0, . . . , 0, 0)
3    U := ∅
4    while u ≺ (1, 1, . . . , 1, 1) do
5    begin
6        if complete(u)
7        then if V := {v ∈ U : u ≺ v} ≠ ∅
8             then U := U ∪ {u} \ V
9             else if    ∄v ∈ U : v ≺ u
10                then U := U ∪ {u}
11        u := next(u)
12    end
13 end
```

Figure 6. The frame of the lexicographic enumeration

5.3 Algorithmic details of the lexicographic enumeration in presence of atoms

5.3.1 The order of coordinates applied in this description.

How to enlarge the algorithm of Fig. 6 if atomic structure of the species is also considered?

First, the lexicographic order is applied. It means that the lexicographically smallest vector is $(0, 0, . . . , 0, 0, 0)$, the second smallest is $(0, 0, . . . , 0, 0, 1)$, the third smallest is $(0, 0, . . . , 0, 1, 0)$, etc.

The dimension of the vector, i.e. M, is the number of species in our case. Index m identifies the species X(m). The order of the coordinates is important from algorithmic point of view.

5.3.2 Checking the atomic condition (A).

Let $\mathcal{A}$ be the set of the different atoms in the chemical species. Similarly, let $\mathcal{M}$ be the set of species. The presence of atom a is described by a characteristic vector as follows. The vector

$$c^a \in \{0, 1\}^M$$

is the characteristic vector of the atom $a \in \mathcal{A}$ if its components are defined as follows:

$$c_m^a = \begin{cases} 1, & \text{if atom } a \text{ is present in species } X(m) \\ 0 & \text{otherwise.} \end{cases}$$

These vectors are fixed during the whole process. Let $L^a := \max\{m|c_m^a = 1\}$, i.e. L^a is the last species to contain atom a. Let u be the characteristic vector of the initial set under construction, i.e. if $u_m = 1$, then species $X(m)$ is present in the set, otherwise 0. An atom is represented in the set, if there is a species in the set $\{X(m); u_m = 1\}$ such that it contains atom a, or $c_m^a = 1$. Hence, an atom is represented in the set if and only if

$$\sum_{m=1}^{M} c_m^a u_m \geq 1.$$

If this inequality is not satisfied, then the equation

$$\sum_{m=1}^{M} c_m^a u_m = 0$$

holds implying $u_{L^a} = 0$. Then, species $X(L^a)$ is put into the initial set, i.e. u_{L^a} is changed to 1. The result is the lexicographically minimal set which contains all the species being present at the beginning of the construction of the current suggestion for the initial set and contains atom a as well. These steps are done for all kinds of atoms.

5.3.3 Checking the minimality condition (B).

Assume that the set of the characteristic vectors of the minimal sets is set U. A smaller than u initial set is contained in U if and only if there is a vector $v \in U$ such that $v \not\succ u$. Notice, that the two vectors cannot be equal as vector u is lexicographically strictly increasing during the process. In other words, this condition means that $v_m = 1$ implies $u_m = 1$, or equivalently

$$\sum_{m=1}^{M} v_m u_m = \sum_{m=1}^{M} v_m.$$

If the condition is satisfied, then the current candidate cannot be minimal. Assume that max1 is the index of the last component having value 1 in the current candidate vector u. Further on, let max0 be the index of the last 0 component in u before the max1 component. Thus

$$\text{max1} = \max\{m|u_m = 1\} \text{ and } \text{max0} = \max\{m|m < \text{max1}, u_m = 0\}.$$

Thus, the end of the candidate vector is $01\ldots10\ldots0$. If any 0 component of the last segment of the vector consisting of only 0s is changed from 0 to 1, then the non-minimality condition still holds. Thus, component max0 must be changed from 0 to 1 as the lexicographically minimal attempt to get out from the non-minimality state. All components behind this component must be 0. It is one of the ways how lexicographic order can enumerate many cases in an implicit way.

5.3.4   The order of the components.

Notice, that $u_m = 1$ in every second iteration in the case of a complete enumeration. Therefore, it is advantageous if it represents a species which is present in a minimal set with high probability. On the other hand, $u_1 = 0$ in the first half of the complete enumeration. Thus, if any minimal set exists containing the first species, it can be obtained only in the second half of the enumeration. To have many minimal sets can accelerate the enumeration discussed in the previous sections. Thus, if there is any available information on the probability of the presence of the species in minimal initial sets, it is worth to change the order of the species accordingly.

5.3.5   Algorithmic summary.

Here is a short algorithmic summary of the discussed procedure. The Boolean variable Volpert-able is true if the current candidate vector satisfies the necessary conditions (A), and (B) and the construction of the Volpert-graph may begin. No new notation is introduced. Relation = means logical condition while := means assignment, as usual.

5.4   Results

Several well known problems were solved using the proposed lexicographic enumeration method to analyse the efficiency of the algorithm. The test problems were solved on a laptop with 8 GB RAM and Intel® Core™ i7-8650U CPU 1.90 GHz 2.11 GHz running a 64-bit Windows 10 operating system. The problems solved consisted of hydrogen combustion models [22] and Leeds3 [26], methanol combustion models, such as Klippenstein2011 [12], Li2007 [14] and Rasmussen2008 [23]. (See also [29] for a collection of combustion models and the references therein.) The specifics of the problems such as total number of molecules, atoms and interactions alongside the total CPU times are reported in Table 3.

```
1 Volpert-able:=false
2 while not(Volpert-able) do
3 begin
4    for a ∈ A do
5       if ∑ⁿₘ₌₁ cᵃₘuₘ = 0 then u_Lᵃ := 1
6    Volpert-able:=true
7    for v ∈ U do
8       if v ⊀ u then
9       begin
10         Volpert-able:=false
11         max1 := {m | uₘ = 1}
12         max0 >= {mm < max1, uₘ = 0}
13         u_max0 := 1
14         for m := max0 to max1 do uₘ := 0
15      end
16 end
```

Figure 7. Summary of the algorithm

Ref. no.	Species	Reaction steps	Atoms	Inter- mediates	CPU time
[32]	8	40	2	4	0.012
[12]	18	172	3	8	0.029
[26]	9	44	2	4	0.014
[14]	18	170	3	8	0.39
[23]	28	320	4	13	26.361

Table 3. Results of the smart method on a few large models

The algorithm fails to solve problems with large number of elements in an acceptable time. The reason can be easily attributed to the large set of possible binary vectors which amounts to 2^n, where n is the total number of species excluding the intermediate ones. As an instance, the problem in Ref. [31] contains 58 elements from which 37 elements are intermediates, which leaves the algorithms 21 elements to deal with. Thus, cardinality of the set of possible lexicographic vectors is $2^{21} = 2,097,152$. Although this number is not that large, it takes more than 72 hours to solve this problem. One of the reasons is the time it takes for each vector to transform and exclude the intermediate elements. Consequently, one of the solutions lies in the first step of the solution method which is the coding of chemical species into numbers. If species other than intermediates are coded first, the lexicographic vector can be divided into two parts; first part describes the existence of main species and the second part which describes the intermediate ones. While calculating the next lexicographic order, the algorithm can easily dispose of the second part and all the

related transformations which can effectively speed up the algorithm. The second solution for large problems is limiting the search space to minimal sets with maximum cardinality of n. For properly small n, the cardinality of the set of possible lexicographic vectors will reduce to $c(m, n)$, where m is the number of non-intermediate elements. The results using the suggestions above is described in what follows. By limiting the cardinality of minimal sets to 5, the problem of Zabetta and Hupa [31] was solved in just 780.49 seconds and 122 minimal sets were found.

The problem in Aranda et al. [1] with 1063 reactions and 76 molecules from which 40 were intermediates was solved within 10729.79 seconds. The maximum cardinality of minimal sets was set to 5 and 434 minimal sets were found.

6 Conclusions and perspectives

6.1 Possible practical applications

An obvious medical application of the solution of the problem is as follows. Suppose an organism should have all the species in a part of the metabolism, and one has to provide initial species to ensure this. Then, we have to choose between the minimal initial species sets in such a way that we provide those species which cause the smallest side-effects, which are easiest to obtain, which are the cheapest, which has the shortest absorption time, etc.

In the case of chemical engineering applications (of which combustion is just one of the possible areas) one of the viewpoints can again be economics or availability, but it may also happen that starting from one minimal set has smaller unwanted effect on the environment.

6.2 Theoretical connections

Let us start from an appropriate minimal initial set. Then all the species will have positive concentrations for all finite (positive) times, or, the trajectory lies in the open first orthant of the species space. In a large class of reaction networks (for weakly reversible networks) this property (permanence) is conjectured to imply ([3, 4]) that the same property holds for "infinite time", as well.

6.3   Further problems

1. An extension for the case of reaction-diffusion models seems to be possible based on the papers by [30] and the more recent ones by [19–21]. Exclusion of cross diffusion would not be a real restriction, if necessary from the mathematical point of view.

2. In our Copenhagen lecture [27] we presented the following theorem.

   Theorem 2. Given a kinetic differential equation choose a few coordinates of the stationary point which is zero, put them into the equations. What you get will also be a kinetic differential equation.

   Based on it one could select some coordinates to be zero, and construct appropriate inducing reactions to the remaining smaller kinetic differential equation, and apply one of the structural conditions to say something about the remaining coordinates if they are positive or zero.

3. Beyond the Theorem 2 above one might also study **siphons**, starting with [25] and continuing with recent developments.

4. Which are the states which can be reached by a reaction at all? This is connected to controllability, and also to the range of the right hand side etc. [28, p. 164].

5. Beyond kinetics: given a polynomial differential equation we may be able to answer the analogous question, if we are able to find a reaction network inducing the give differential equation. Ref. [6] provides answers to questions of this last type.

References

[1] V. Aranda, J. M. Christensen, M. U. Alzueta, P. Glarborg, S. Gersen, Y. Gao, P. Marshall, Experimental and kinetic modeling study of methanol ignition and oxidation at high pressure, *Int. J. Chem. Kinet.* **45** (2013) 283–294.

[2] M. Banaji, G. Craciun, Graph-theoretic approaches to injectivity and multiple equilibria in systems of interacting elements, *Commun. Math. Sci.* **7** (2009) 867–900.

[3] J. D Brunner, G. Craciun, Robust persistence and permanence of polynomial and power law dynamical systems, *SIAM J. Appl. Math.* **78** (2018) 801–825.

[4] G. Craciun, F. Nazarov, C. Pantea, Persistence and permanence of mass-action and power-law dynamical systems, *SIAM J. Appl. Math.* **73** (2013) 305–329.

[5] G. Craciun, C. Pantea, E. D. Sontag, Graph–theoretic analysis of multistability and monotonicity for biochemical reaction networks, in: H. Koeppl, G. Setti, M. di Bernardo, D. Densmore (Eds.), *Design and Analysis of Biomolecular Circuits*, Springer, 2011, pp. 63–72.

[6] G. Craciun, G. Szederkényi, M. D. Johnston, E. Tonello, J. Tóth, P. Yu, Realizations of kinetic differential equations, *Math. Biosci. and Engng.* **17** (2020) 862–892.

[7] P. Donnell, M. Banaji, Local and global stability of equilibria for a class of chemical reaction networks, *SIAM J. Appl. Dyn. Sys.* **12** (2013) 899–920. .

[8] M. Feinberg, *Foundations of Chemical Reaction Network Theory*, Springer, 2019.

[9] E. Feliu, C. Wiuf, A computational method to preclude multistationarity in networks of interacting species, *Bioinformatics* **29** (2013) 2327–2334.

[10] F. Friedler, K. Tarján, Y. W. Huang, L. T. Fan, Graph–theoretic approach to process synthesis: axioms and theorems, *Chem. Eng. Sci.* **47** (1992) 1973–1988.

[11] A. Gábor, K. M. Hangos, G. Szederkényi, Linear conjugacy in biochemical reaction networks with rational reaction rates, *J. Math. Chem.* **54** (2016) 1658–1676.

[12] S. J. Klippenstein, L. B. Harding, M. J. Davis, A. S. Tomlin, T. Skodje, Uncertainty driven theoretical kinetics studies for CH_3OH ignition: $HO_2 + CH_3OH$ and $O_2 + CH_3OH$, *Proc. Comb. Inst.* **33** (2011) 351–357.

[13] K. Kovács, B. Vizvári, M. Riedel, J. Tóth, Computer assisted study of the mechanism of the permanganate/oxalic acid reaction, *Phys. Chem. Chem. Phys.* **6** (2004) 1236–1242.

[14] J. Li, Z. Zhao, A. Kazakov, M. Chaos, F. L. Dryer, J. J. J. Scire, A comprehensive kinetic mechanism for CO, CH$_2$O, and CH$_3$OH combustion, *Int. J. Chem. Kinet.* **39** (2007) 109–136.

[15] D. Lichtblau, Symbolic analysis of multiple steady states in a MAPK chemical reaction network, *J. Symb. Comp.*, submitted.

[16] T. M. Liggett, *Continuous Time Markov Processes: An Introduction*, AMS, Providence, 2010.

[17] L. Lovász, *Combinatorial Problems and Exercises*, AMS, Providence, 2007.

[18] N. I. Markevich, J. B. Hoek, B. N Kholodenko, Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades, *J. Cell Biol.* **164** (2004) 353–359.

[19] M. Mincheva, M. R. Roussel, Graph-theoretic methods for the analysis of chemical and biochemical networks. I. Multistability and oscillations in ordinary differential equation models, *J. Math. Biol.* **55** (2007) 61–86.

[20] M. Mincheva, D. Siegel, Stability of mass action reaction-diffusion systems, *Nonlin. Anal.* **56** (2004) 1105–1131.

[21] M. Mincheva, D. Siegel, Nonnegativity and positiveness of solutions to reaction–diffusion systems, *J. Math. Chem.* **42** (2007) 1135–1145.

[22] M. Ó Conaire, H. J. Curran, J. M. Simmie, W. J. Pitz, C. K. Westbrook, A comprehensive modeling study of hydrogen oxidation, *Int. J. Chem. Kinet.* **36** (2004) 603–622.

[23] C. L. Rasmussen, J. Hansen, P. Marshall, P. Glarborg, Experimental measurements and kinetic modeling of CO/H$_2$/O$_2$/NO$_x$ conversion, *Int. J. Chem. Kinet.* **40** (2008) 454–480.

[24] V. Reddy, M. Mavrovouniotis, M. Liebman, Petri net representations in metabolic pathways, in: *Proceedings First International Conference on Intelligent Systems for Molecular Biology*, 1993, pp. 328–336.

[25] A. Shiu, B. Sturmfels, Siphons in chemical reaction networks, *Bull. Math. Biol.* **72** (2010) 1448–1463.

[26] G. P. Smith, D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. Th. Bowman, R. K Hanson, S. Song, W. C. Gardiner , V. V. Lissianski, Z. Qin, Gri–mech 3.0, 1999, *http://www.me.berkeley.edu/gri_mech*.

[27] J. Tóth, On the form of kinetic differential equations, Workshop on Mathematical trends in reaction network theory, July 1–3, 2015, Copenhagen, 2015.

[28] J. Tóth, A. L. Nagy, D. Papp, *Reaction Kinetics: Exercises, Programs and Theorems. Mathematica for Deterministic and Stochastic Kinetics*, Springer, New York, 2018.

[29] J. Tóth, A. L. Nagy, I. G. Zsély, Structural analysis of combustion mechanisms, *J. Math. Chem.* **53** (2015) 86–110.

[30] A. I. Volpert, Differential equations on graphs, *Mat. Sb.* **88** (1972) 578–588.

[31] E. C. Zabetta, M. Hupa, A detailed kinetic mechanism including methanol and nitrogen pollutants relevant to the gas-phase combustion and pyrolysis of biomass-derived fuels, *Comb. Flame* **152** (2008) 14–27.

[32] I. G. Zsély, C. Olm, R. Pálvölgyi, T. Varga, T. Nagy, T. Turányi, Comparison of the performance of several recent hydrogen combustion mechanisms, in: *Proceedings of the European Combustion Meeting 2013*, 2013.

7 Appendix: The Code of the Brute Force Solution

This code is for selecting the minimal subsets from the good ones.

```
ProperSubsetQ[x_, y_] := And[SubsetQ[x, y], Not[Equal[x, y]]];
MinimalOnes[l_List] := Fold[Function[{x, y}, Select[x,
Not[ProperSubsetQ[#, y]] &]], l, l]
```

Let us decide if $+\infty$ is among the Volpert index in a reaction with a given set of initial species.

```
FiniteVolpertIndexQ[ini_, reac_] :=
FreeQ[VolpertIndexing[reac, ini], Infinity, 3].
```

It is only the reactant species which are to be considered.

```
ReactantSpecies[reac_] := Union[Cases[Level[Map[ Part[#,1]&,
ReactionsData[{reac}]["fhjgraphedges"]], {-1}],Except[_Integer]].
```

A possible set of intermediates may be

```
inter={Br⁻,BrO₃⁻,CH₃,Cl*,H,H⁺,HO₂,I⁻,IO₃⁻,O,OH⁻, Mn(OH)₂⁺,C}
```

but the default is that there are no intermediates present.

The function `GetElements` gives all the atoms present in a set of species, and also, all the atoms present in a set of reaction steps. Both `GetElements[{H, Cl, HCl}]` and `GetElements[{H + Cl -> HCl}]` give {Cl, H}. Using this function one can easily formulate the code needed.

```
AtomsPresentQ[reac_, subset_] := GetElements[reac] == GetElements[subset]
```

In case if the species have an atomic structure, we take this fact into consideration, otherwise not. An option will be introduced to take care of this. Next, we form the difference of the set of species and the intermediates. In general, this set will be the base set from which to form the power set. In the atomic case we restrict ourselves further to those sets which contain all the atoms present in the reaction network. Then we select those elements of the power set which produce finite Volpert indexes, and finally select the minimal ones among them.

```
Options[MinimalInitials] = {Atomic → False};
MinimalInitials[reac_,intermediates_:{}, OptionsPattern[]]:=Module[{MM,pb},
pb = Complement[reactantSpecies[reac], intermediates];
MM = If[OptionValue[Atomic], Select[Select[Rest[Subsets[pb]],
(GetElements[#] == GetElements[pb])&],
FiniteVolpertIndexQ[#, reac]&],
Select[Rest[Subsets[pb]],FiniteVolpertIndexQ[#,reac]&]];MinimalOnes[MM]]
```

Dealing with the atoms first may strongly reduce the number of the elements which will form the power set. In the case of the Briggs-Rauscher reaction it is only 37% of the subsets that have to be considered, as the result of

```
AtomicSaving[GetReaction["Briggs-Rauscher"]]
```

is 0.372082, using the code

```
AtomicSaving[reac_] := Module[
{ini = RestSubsetsReactionsData[{reac}]["species"], sel,
total = GetElements[{reac}]},
sel = Select[ini, GetElements[#] == total&];
N[Length[sel]/Length[ini]]]
```

The ratio of subsets left in a few reaction can be seen in the Table 4.

Name	Ratio of subsets left
Briggs-Rauscher	0.37
Chapman cycle	0.47
Emanuel-Knorre	0.84
FKN	0.26
Ogg	0.97
Petri	0.22
Vaiman	0.24

Table 4. Saving via utilizing atomic composition