An automated method for graph-based chemical space exploration and transition state finding.

Pablo Ramos Sánchez,[1] Jeremy N. Harvey,[2] and Jose A. Gámez*[1]

Correspondence to: Jose A. Gámez (E-mail: *jose.gamez@covestro.com*)

[1] *Covestro Deutschland AG, Leverkusen, Germany*
[2] *Department of Chemistry, KU Leuven, Leuven, Belgium*

ABSTRACT

Applications of algorithms that automatically explore the chemical space have been limited to chemical systems with a low number of atoms due to expensive involved quantum calculations. Nonetheless, it is possible to explore large regions of the chemical space with low-cost graph theory techniques, to later apply quantum calculations on relevant reaction paths. The method described here tackles the problem of chemical exploration by generating reaction networks with heuristics based on chemical theory. This is done by defining molecular graph transformations that represent elementary reactions in a graph theory approach. Such transformations act upon functional groups in molecules that fulfil the Lewis Structure, which are represented in terms of bond order matrices. This way, a study showing computational time and method's performance in five different chemical systems is presented with a concise chemical representation of graphs, to finally apply an efficient combination of quantum chemical calculations on the reaction paths.

Introduction

A reaction mechanism can be understood as a collection of chemical paths that connect the reactant with a product via relevant molecules, also referred to as reaction intermediates. Finding such a collection of molecules and paths is what is called exploration of chemical space, and can take place digitally by manually changing the molecular structure, based on chemical intuition, and performing subsequent quantum-chemical calculations. Such calculations perform bonding transformations by distorting bonds and bond angles in molecules that represent a reactant or a minimum in the potential energy surface.[1–3] This can be done to locate other minima and saddle-points of the potential energy surface (PES) describing the reaction.[4–8]The saddle-points connect the reactant minimum with other minima that represent possible products or intermediates on the same potential energy surface. Each saddle-point corresponds to an elementary reaction step, and the transition state for this step, and its properties determine the probability of reaction. Although quantum-chemical calculations allow us to understand the relevance of a molecule (or of the chemical path the molecule belongs to) based on energetic criteria, finding relevant paths and performing the calculations requires human expertise and can be computationally expensive.

There are alternative methods that also allow bond transformations without relying on expensive quantum-chemical calculations,[9–20] such as adjacency matrix transformations. [14,21–23] These graph-based transformations can explore large regions of chemical space by adding or subtracting integer values to adjacency matrices. These adjacency matrices can represent molecular species such as a reactant minimum, and the matrix resulting from the transformation can represent intermediates or a product minimum on the PES. However, as such an approach does not involve quantum-chemical calculations, saddle-points cannot be found, so important information regarding the reactivity of the chemical species, and therefore the relevance of the chemical paths, is not included. Besides, chemical space is still very large so even while using adjacency matrices, exploring it completely for a given supramolecular system (normally the reactant) is challenging, especially when considering systems with a large number of atoms.[24–26] Therefore, calculating a reaction network that shows every connected minimum for the given system becomes normally an unfeasible task. This problem, together with the computational time and human expertise that finding saddle-points requires, has motivated several groups to develop automatic data-driven algorithms. [27–41] These algorithms attempt to narrow the chemical space to explore in such a way that only important intermediates and transition states located between the reactant and a given product are considered.

The following provides a non-exhaustive overview of efficient automated chemical space exploration methods that have been shaping the landscape to date:

Chemical space exploration methods based on graph theory are 1) the ACE-Reaction[14,42] developed by W. Y. Kim et al., which creates a chemical network by applying bond-addition and bond-breaking matrix transformations from randomly initialized matrices, to later computing relevant transition states; or 2) the Automatic Proposal of Multistep Reaction Mechanisms approach proposed by Habershon et al.[21] that uses reaction class transformation matrices to specifically look for a product. While the first relies on stochastic procedures that can lead to a combinatory explosion, the second relies on a limited pre-defined database and cannot be applied to those reaction families for which data is not available.

Efficient algorithms based on pure ab initio quantum-chemical calculations are 1) The Zstruct2 code from the Zimmerman Group[43,44], that combines the single-ended GSM reaction pathfinder[45,46] with bond-addition or bond-breaking vectors that describe elementary reactions, and that has been applied to transition metal-catalyzed reactions and polymerization;[47–51] and 2) the Nanoreactor tool developed by Martínez-Núñez et al. that performs fast high-temperature and pressure ab-initio molecular dynamics to find minima,[38,52] and that has successfully been applied in reactions such as acetylene polymerization, in which chains of more than 70 atoms were grown.[53] Computational times needed for ab initio calculations are nonetheless larger than the ones derived from a graph theory approach when generating a reaction network.

To summarizing, when describing chemical reactions, methods that rely on pure ab initio quantum chemical calculations can efficiently find transition states, but these are usually limited to reactions in which reactants and products are connected by a small number of elementary steps. Otherwise, they will require a large amount of computational time. There are many situations in which the number of elementary steps is large. Some examples of such reactions are the metabolic reactions, such as the glycolysis that can be described in 10 elementary steps with systems up to 25 atoms. Although promising, current algorithms still lead to a combinatorial explosion due to the large number of possible reactions that can be described in the reaction network, making it unfeasible to later study every single elementary step with expensive ab initio quantum chemical calculations.

This is why, in this publication, we focus on tackling the general problem related to the size of the chemical space by first decreasing the amount of time that network generation demands, to later combine ab initio quantum calculations in a fashion way. Therefore, our approach also relies on chemical exploration with graph-theoretical techniques to 1) quickly find chemical species (nodes in the network) and generate connections between them that represent elementary transformations from a graph-theoretical perspective (edges in the network), and 2) fully automate transition state search for every couple of connected nodes in the network. The first step is carried out by defining molecular graph transformations that represent elementary reactions in a graph-theoretical approach. Such transformations act upon functional groups in molecules whose Lewis structure satisfies the octet rule. These molecules are represented in terms of bond order matrices, which are related to adjacency matrices but also contain information about bond orders. The second step is performed using the double-ended GSM reaction pathfinder for different conformers in combination with an automated multi-step reaction detector, with refinement and intrinsic reaction coordinate (IRC) validation of transition states.

The methodology followed will be explained in the Methods section. It has been structured into two parts. The first one englobes the reaction network generation. This part will begin with the definition of the chemical transformation of bond order matrices, followed by the reaction network generation procedure and reduction of the network. The second part englobes the quantum-chemical calculations. This part contains the entire quantum flowchart: from the generation of conformers to the validation of transition states. After the Methods section, we will show the results obtained by applying our method to five chemical systems. We will conclude with future applications and next steps in the development of chemical explorations.

Methods

Reaction Network Generation

Bond Order Matrix Transformation

Supramolecular systems formed by one or more molecules can be represented in terms of connectivity graphs. These are square matrices *N* x *N,* where *N* is the number of atoms in the system. The off-diagonal elements l_{jk} can take the values 1 or 0 depending on whether or not the j^{th} and k^{th} atoms are bonded to each other. This matrix representation of connectivity is often referred to as the adjacency matrix of the system or the atomic matrix).

To describe a chemical transformation, one needs to specify an operation upon the adjacency matrix $\mathbf{C^0}$ that converts it into the matrix for another species. This can be done by defining a transformation matrix $\mathbf{T^0}$, most of whose elements are equal to zero, with a smaller number of off-diagonal elements corresponding to changes in connectivity. In previous studies, these non-zero elements have been set randomly (with some minor constraints),[14,42] or by using templates, to be either −1 or 1.[21] Summing $\mathbf{T^0}$ and $\mathbf{C^0}$ gives $\mathbf{C^1}$, which may or may not map onto a supramolecular species that corresponds to a valid Lewis Structure. Upon repeatedly carrying out the same process, thereby

generating successive families of $\mathbf{T}^n$ and $\mathbf{C}^n$ matrices, one can perform an exhaustive exploration of all possible species for the given atomic composition, but this exhaustive exploration also leads to a combinatory explosion in the total number of possible adjacency matrices, which increases very steeply as the number of atoms increases.

To describe the bond order of molecules, a transformation in the adjacency matrix that takes into account valence rules must be performed. The resultant matrix is called a bond order matrix. These are also *N* x *N* square matrices but the off-diagonal $\mathbf{b}_{jk}$ elements now take integer values that give the bond order associated with atoms j and k (0 represents no bond, 1 represents a single bond, 2 represents a double bond, etc.).

Our approach removes the idea of working in the entire adjacency matrix space because of its large dimensionality to avoid a combinatory explosion. Instead, we directly work in the chemical space corresponding to structures following the octet rule. Restriction to this space can be enforced when performing a transformation over a bond order matrix $\mathbf{b}^0$ of a given supramolecule, by using the following approach:

We first select two bonds in the molecule which will be broken: $bond_{jk}$ and $bond_{j'k'}$, from which we get the pairs of indices *j* and *k*; and *j'* and *k'* which correspond to linked atoms respectively, that denote the three or four atoms involved in the reaction step (three is the minimum number of atoms that is frequently involved in an elementary transformation). Two different product bonding combinations can take place by shuffling the indices involved in bonds while respecting valence rules: one can create either 1) $bond_{jj'}$ and $bond_{kk'}$ or 2) $bond_{jk'}$ and $bond_{kj'}$, leading to two new bond order matrices. This is done for every pair of bonds in the molecule.

This sort of *two-bonds-breaking-two-bonds-forming* matrix transformation applied over a bond order matrix of a molecule that satisfies the octet rule corresponds to an elementary transformation from a graph-theoretical perspective (and it, therefore, generates a supramolecule that also satisfies the octet rule). We believe that this approach is equivalent to using the combination of addition and dissociation reaction classes as in Habershon et al.'s work.[21] While the present transformations are unable to create ionic species (whose formation requires a transformation in which a different number of bonds are broken and formed for a given atom: two-bonds-breaking-one-bond-forming, one-bond-breaking-two-bonds-forming, etc.), the use of such more specific bond transformations does have the effect of highly reducing the size of the chemical space that needs to be explored.

The three described approaches are illustrated in **Figure 1**, which shows how (a) Kim's method[14] and (b) Habershon's method[21] (red background) describe bond-breaking and bond formation applied over adjacency matrices, relying in the first case on exhaustive random matrix generation and the second case on reaction class heuristics. Both methods later require a second algorithm that transforms the adjacency matrix into a possible bond order matrix (shown with a green background). Because of the large size of the adjacency matrix space, most generated matrices do not map onto a valid bond order matrix, while application of a *two-bonds-breaking-two-bonds-forming* transformation to a valid bond order matrix as shown in (c) (blue background) is guaranteed to lead to a bond order matrix that fulfils the octet rule, without requiring additional valence rules.

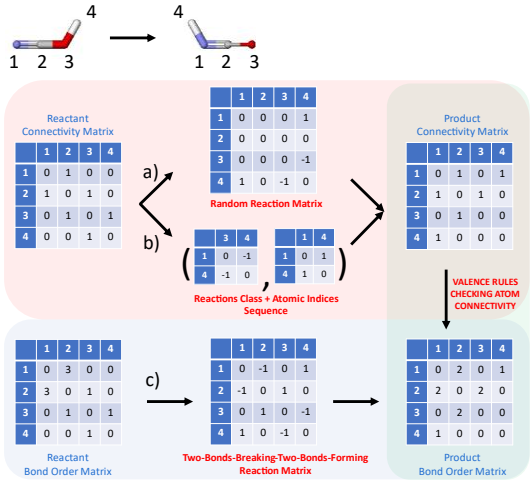

Figure 1. Three methods for defining chemical reactions with matrix transformations, as illustrated for the four-atom cyanic acid – isocyanic acid system. (a) Definition of a randomly generated N x N reaction matrix applied over the cyanic acid adjacency matrix. (b) Definition of a sequence of 2 x 2 reaction classes applied over the cyanic acid adjacency Matrix. (c) Definition of the Two-Bonds-Breaking-Two-Bonds-Forming Reaction Matrix applied over the cyanic acid Bond Order Matrix.

Exploration of chemical space with two-bonds-breaking-two-bonds-forming processes

Given this property of *two-bonds-breaking-two-bonds-forming* matrix transformations, and the drastically reduced size of chemical space for bond-order matrices that satisfy the octet rule, a combinatory explosion is less severe and we find our procedure can survey the entire available non-ionic chemical space without needing to rely on stochastic procedures.

Nonetheless, the chemical space is still very large. To reduce the number of possible combinations in a chemically intuitive way, a simple database with functional groups can be automatically generated, and some of these groups can be held fixed during the procedure so that the number of bonds that are allowed to transform is also drastically reduced. This implies a reduction in the bond order dimensionality from *N* x *N* to *M* x *M* where atoms in the reduced matrix representation are treated as a group of non-separable atoms that belong to detected functional groups. These atoms are also referred to as active atoms. This way, two-bonds-breaking-two-bonds-forming transformations applied over the so defined active atoms represent in a chemical sense functional group transformations. **Figure 2** shows an example of functional group detection and posterior matrix reduction to its active atoms representation applied to methanolamine.

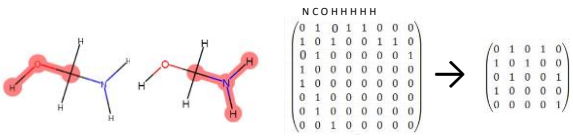

Figure 2. Functional groups detection of methanolamine and matrix reduction.

In the same way, a second database was developed and used to prune unwanted molecules from the chemical space to be explored. For example, for reactions of typical organic compounds, with their large number of hydrogen atoms, stochastic exploration of bond order matrices has a high probability of generating species that include molecular hydrogen (whose formation is not energetically favorable compared to organic reactions). Bond-order matrix transformations leading to these species can be removed from possible steps, leading to a significant reduction in the size of space that needs to be explored. More details of these databases can be found in the supporting information (SI).

Once a set $\{\mathbf{b}\}^1$ is created from $\mathbf{b}^{reac}$ we already know that every molecule in $\{\mathbf{b}\}^1$ is connected to $\mathbf{b}^{reac}$ by an two-bonds-breaking-two-bonds-forming transformation, so we include those connections in the chemical network provided that the molecular mechanics energy of the new species is not prohibitively high. Recursively repeating the whole process using the new species in $\{\mathbf{b}\}^{m+1}$ as starting points, we can generate our chemical network. Previous methods that work in the adjacency matrix space need a collection of adjacency matrix transformations to represent a single elementary reaction, a second algorithm that transforms adjacency matrices into bond order matrices, and a third that connects bond order matrices to form a reaction network [14,21,42]. Our method skips these three steps by creating the network on the fly. This brings the next advantages compared to previous methods:

1. A more concise and simpler implementation of the method
2. Since the method works in a reduced version of the space, every possible combination is considered in such space
3. Avoidance of non-chemical intermediates generation, with the corresponding time reduction of exploration
4. Direct connection of all intermediates found

As a consequence, the chemical space is drastically reduced to a more meaningful region. **Figure 3** provides a qualitative depiction of reaction networks generated with different methods.

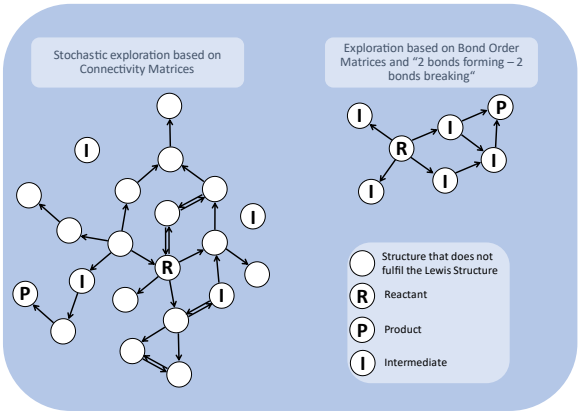

Figure 3. Reaction networks are generated with different algorithms. (Left) Reaction network generated in a stochastic way: node transformations are defined as random reaction matrices acting on adjacency matrices. (Right) Reaction network generated by performing *two-bonds-breaking-two-bonds-forming* reaction matrix transformations that act over the bond order matrix space.

Growth of the reaction network

The reaction network can be understood from a graph-theoretical perspective. Different chemical species represent nodes in the network. These are connected by edges representing elementary reactions, loosely defined. The method here described shows two ways of exploring the chemical space or making the reaction network grow: branched growth and layered growth.

In branched growth, the network is initialized with the set $\{\mathbf{b}\}^0$ containing just the reactant bond order matrix $\mathbf{b}^{reac}$. A set of expansion cycles on the sets $\{\mathbf{b}\}^n$ then follow, in which a random element $\mathbf{b_j}$ **from** $\{\mathbf{b}\}^n$ is acted upon by the transformation matrix $\{\mathcal{R}\}_i$ yielding a set of new elements $\{\mathcal{R}_i\mathbf{b}_j\}$ that belong to $\{\mathbf{b}\}^{n+1}$. Repeating the process creates a branched structure:

$$\text{iter} = 0: \quad \{\mathbf{b}\}^0 = \{\mathbf{b}^{reac}\}$$
$$\text{iter} = 1: \quad \{\mathbf{b}\}^1 = \{\mathbf{b}^{reac}\} \cup \{\mathcal{R}_i\mathbf{b}^{reac}\}$$
$$\text{iter} = 2: \quad \{\mathbf{b}\}^2 = \{\mathbf{b}\}^1 \cup \{\mathcal{R}_i\mathbf{b}_\text{j} \mid (\mathbf{b}_\text{j} \in \{\boldsymbol{b}\}^{\mathbf{1}}) \text{ for some j}\}$$
$$\ldots$$
$$\text{iter} = N: \quad \{\mathbf{b}\}^N = \{\mathbf{b}\}^{N-1} \cup \{\mathcal{R}_i\mathbf{b}_\text{j} \mid (\mathbf{b}_\text{j} \in \{\boldsymbol{b}\}^{\boldsymbol{N-1}}) \text{ for some j}\}$$

In layered growth, having created a set $\{\mathbf{b}\}^1$ from $\mathbf{b}^{reac}$ that represents the set of nodes connected by one elementary transformation from the reactant, every element in $\{\mathbf{b}\}^1$ is transformed before starting to generate a second 'layer' of elements $\{\mathbf{b}\}^2$ by transforming members of $\{\mathbf{b}\}^1$. Once every element in $\{\mathbf{b}\}^1$ has been transformed, the network contains the set of edges that represent the first and second layers of elementary reactions from the reactant. The process is repeated for more iteration cycles until the desired product is found or after one has generated a given number N of growth layers. This way, the reaction network grows by layers, every layer *i* representing

the set of species connected by *n* elementary reactions (or more precisely two-bonds-breaking-two-bonds-forming transformations) to the reactant. For simplicity, we will refer to the layers that describe elementary reactions from reactant as "elementary layers from the reactant".
The procedure is the same as in branched growth but applies transformation matrices for all j instead of some j at any iteration cycle.

While we believe that the branched growth strategy is useful to generate a database of molecules and transition states when a specific product is not required, the layered growth strategy allows us to reduce considerably the time of calculation when information about a product is provided: if reactant and product are connected by several elementary reactions, working layer by layer ensures that the algorithm does not exceed the needed number of elementary reactions. Therefore, the layered growth strategy is especially useful for large chemical spaces in which a single node connects to dozens of them and branching leads to a combinatory explosion.

Network Reduction:
Once the chemical space has been explored and reactant and product are connected, the number of nodes in the network can still be very high depending on the number of atoms and the number of elementary layers relative to the reactant that has been explored. Since expensive calculations of transition states are later performed, it can be desirable to reduce the network to just consider a more relevant fraction of nodes. Several algorithms are implemented to achieve this: Breadth-first search (BFS),[54] combined with a network reduction strategy, and Dijkstra's algorithm.[55]

In the BFS approach, the first step is to calculate the chemical distance (minimum number of edges that connect two given nodes) between the reactant R and the product P, this *is* $D_{R,P}$; then the matrix *D* containing the chemical distances between each pair of nodes in the network is computed.

We then apply the next inequalities to generate a reduced network:

$$\delta_i = \begin{cases} 0 & \text{if } D_{i,R} + D_{i,P} \geq D_{R,P} + d_{input} \\ 1 & \text{otherwise} \end{cases}$$

If $\delta_i = 1$, node *i* is included in the reduced network. With the smallest possible value of $d_{input} = 0$ (default value), this criterion means that only the nodes situated along a pathway leading from reactant and product with a minimal number of edges $D_{R,P}$ are kept in the reduced network. Larger values of d_{input} lead to including nodes that lie outside these minimum-edge pathways. This inequality has also been previously used in previous work.[14,42] One advantage of this definition is that it does not assume the validity of the heuristic "principle of minimum chemical distance" (PMCD), whereby the minimum energy path is considered to be one whose number of nodes is the lowest. This is only assumed in case that $d_{input} = 0$. For larger values of this parameter, we are also considering paths that connect reactant and product through a larger number of nodes. The control over the degree of reduction of the network provided by choosing the parameter d_{input} becomes especially useful when several products are generated in a reaction, and someone is interested in knowing how many such species are possible.

Nonetheless, this strategy applied over networks with hundreds or thousands of nodes can still provide very large reduced networks. For these scenarios, Dijkstra's algorithm[55] can be used over the already reduced matrix. This algorithm will only extract paths that follow the PMCD. However, this only becomes useful when someone is looking for specific energy barriers since this principle has been proven false numerous times when applying quantum chemistry calculations.

The database of functional groups can also serve as a strategy for reducing the network: if the user is interested in a specific reactivity, by only considering functional groups that describe it, a more guided exploration can take place with the consequent reduction in the network size. This nonetheless requires some basic chemical expertise, and in case the database does not contain all relevant functional groups can lead to incorrect mechanisms. We believe that this is the first reduction strategy based on chemical reactivity.

To maximize optimal network generation, we will see in the Results section that the layered growth strategy and functional groups databases have been used for every system under study.

Exploration of the PES

Conformer generation
Once a network is created, transition states are calculated for every two connected nodes $\mathbf{i}, \mathbf{j}$ in the reaction network. To do this, a 3D geometrical representation of $\mathbf{i}, \mathbf{j}$ must first be chosen. Such an arrangement of atoms and molecules with specific coordinates in 3D space, thereby involving spatial interactions between all atoms in the system, even those that are not connected through the chemical bonding network, is what we refer to as a conformer. Generating coordinates for a conformer requires taking into account preferred bonding structures as well as minimizing sterical repulsion forces between groups within one molecule or between groups that are part of different molecules. To do this, one usually must start from randomly initialized atom coordinates slightly constrained to empirical bond and angle values. This however does not ensure that the minimized reactant and product conformers are connected by the minimum energy path. To maximize the probability of finding the minimum energy path in which the transition state is located, sets of conformers $\{\mathbf{i}\}$, $\{\mathbf{j}\}$ are generated.

The next steps are followed using the 3rd version of the ETKDG algorithm as implemented in the RDKit toolset[56]:

1. Depending on the number of atoms, hundreds or thousands of conformers were randomly generated for a given species that represents the reactant of the reaction **i**, **j** in the network.
2. Butina's clusterization is performed over the list of conformers. The criteria to follow during the clusterization procedure is the maximization of the RMSD. This way, a set of conformers $\{\mathbf{i}\}$ is extracted from the total number of initially generated conformers**.** The elements of $\{\mathbf{i}\}$ are representatives of the cluster (also referred to as centroids), and the RMSD is maximum between them.
3. Given a conformer **i** from $\{\mathbf{i}\}$ that represents the reactant R, a conformer **j** that represents the product P is created by breaking and forming the appropriate bonds in conformer **i** and then optimizing the energy of the resulting structure of the species of connectivity corresponding to **j** at the molecular mechanics level of theory. By generating conformer **j** from conformer **i** we ensure that atom labeling in both conformers is constant. This is an important condition to be fulfilled when later applying the GSM.

Quantum Calculations

Once a chemical network is created, we compute the activation energy for every reaction described by two connected nodes. The activation energy is related to the rate constant for the associated step, which in turn can be used to estimate a weight associated with the corresponding edge in the network. For this purpose, we can find the minimum energy path that connects a reactant with a product by evaluating weights along different paths.
Nonetheless, caution must be used when interpreting the edges in the chemical network – which are elementary transformations from a graph-theoretical perspective – as elementary steps in a quantum-chemical sense. In fact, such transformations may not be elementary in the quantum-chemical sense, such that one may instead find a non-elementary reaction path involving multiple maxima and with intermediate formation of new chemical species that had not been predicted when applying graph theory techniques. For such cases, it is important to identify the collection of elementary steps that lead to the minimum activation energy, which can be done by analyzing the energy profiles derived from quantum calculations.

Activation energies are computed by finding saddle points on the potential energy surface, which we will refer to here as transition states. Due to the computational cost that is involved when finding the structure of transition states, a strategy of mixing different levels of theory is described below.

Given the conformers $\mathbf{i}, \boldsymbol{j}$ from $\{\mathbf{i}, \mathbf{j}\}$:

1. Structural optimization of **i** and **j** is performed using an efficient semiempirical *GFN2-xTB* level of theory;[57]
2. The Growing String Method (*GSM*) is used to find the TS connecting **i** and **j**, again using the faster *GFN2-xTB* level of theory;
3. The generated energy path is analyzed to check if the energy profile corresponds or not to an elementary reaction, being categorized as elementary if only one maximum connecting two minima that represent different chemical species is found. This can be done by first automatically identifying minima based on the energy gradient, then converting their atom coordinates to SMILES format with Pybel[58], and finally comparing the SMILES of every two consecutive minima. If it is the case that the energy profile describes an elementary reaction, one proceeds to the next step; if not, one returns to step 1 but selects a different conformer pair from $\{\mathbf{i}, \mathbf{j}\}$**.**
4. A single energy point calculation is performed at the structure of the transition state on the semiempirical computed *GSM* path using a higher level of theory, namely the DFT functional B3LYP[59]*,* leading to the energy E.

Repeating steps 1, 2 and 3 for the different elements of $\{\mathbf{i}, \mathbf{j}\}$ leads to a set of transition state energies $\{\boldsymbol{E_{ij}^{TS}}\}$ of the involved species ***i*** and ***j***. The transition state with the lowest energy $\boldsymbol{E_{lowest}^{TS}}$ from this set $\{\boldsymbol{E_{ij}^{TS}}\}$ is then refined and validated by performing a new structural optimization, but this time at a higher level of theory (here, this higher level of theory was the DFT B3LYP[59] method with the 6-31G(d, p) basis set and Grimme's dispersion correction terms D3 with Becke-Jones (BJ) damping.[60] This is done with an in-house software named *SubautoTS*[61] that moves the geometry closer to the saddle-point on the potential energy surface with a driven Monte Carlo algorithm, and posterior validation by evaluating the intrinsic reaction coordinate.

Results and Discussion

Reaction network generation with graph theory

To validate our method's performance, five molecular systems as presented in Figure 4 were studied. Canonical SMILES input format[62] that represents the reactant and the product is provided. Systems differ in their complexity regarding the number of atoms, atom types, and the maximum number of elementary reactions that are needed to connect the reactant with a product.

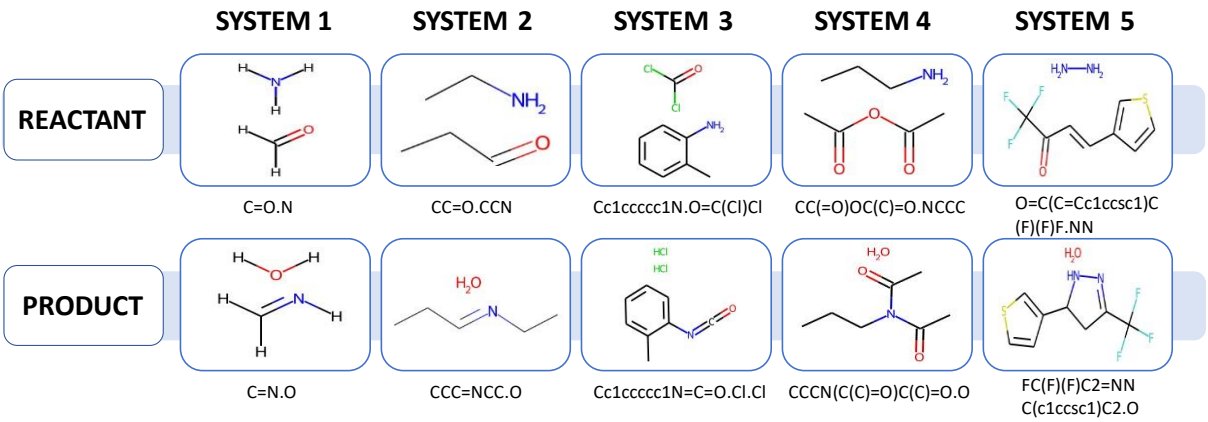

Figure 4. Studied systems

Table 1. Chemical systems studied with their corresponding: SMILES, network sizes, number of atoms, number of elementary reactions, and CPU time required for the network generation with the layered growth method

System	Number of atoms	Number of elementary reactions between R and P	Network size	Reduced network size with BFS $d_{input} = 1$	Reduced Network size with Dijkstra	CPU time (hours:minutes:seconds)
1	8	2	5	5	3	00:00:03
2	20	2	26	9	3	00:00:21
3	21	2	35	11	3	00:00:51
4	26	2	197	37	3	00:08:12
5	24	3	1185	189	4	02:57:50

Table 1 shows how the computational time needed for network generation increases with the number of nodes in the network (network size) when the layered growth method, the complete database of functional groups, and the database of non-desired molecules (databases provided in the SI) are applied. The network size increases with the number of atoms and the number of elementary reactions that connect reactant and product in the system. By comparing the network size between systems 4 and 5 we show that the number of nodes especially grows with the number of elementary reactions. We also find that network

reduction with the BFS approach[54] can be performed without thereby eliminating important reaction paths between reactant and products provided that d_{input} is high enough. However, for chemical systems of larger size (such as the present system 5), even after reduction with the BFS approach, the reduced networks still contain a large number of reactions to be evaluated with the subsequent quantum chemistry calculations. To avoid this problem, a second reduction can be performed with Dijkstra's algorithm.[55] In a reaction network, a path with the minimum number of nodes does not necessarily coincide with the minimum energy path, and although Dijkstra's algorithm reduces the network size to only keep those paths that connect reactant and product through the minimum number of nodes, it can lead to neglect of the minimum energy path connecting the reactant with a product. Related to this issue, by definition, the use of a catalyst is known to decrease the overall energy barrier of a reaction, but it usually does this by introducing additional elementary steps, for example, reactions between the reactant and a catalyst, to the process. Hence a reaction path involving a catalyst will almost inevitably take a larger number of elementary steps to go from reactants to products than a catalyst-free pathway that fulfils the PMCD. In fact, even without explicitly adding extra molecules that work as catalysts, minimum energy pathways from reactant to product may not satisfy the PMCD. To illustrate this, we show an example of the reaction from methanolamine to methylhydroxylamine, NCO → CNO, found in system 1, illustrated in **Figure *5*.**

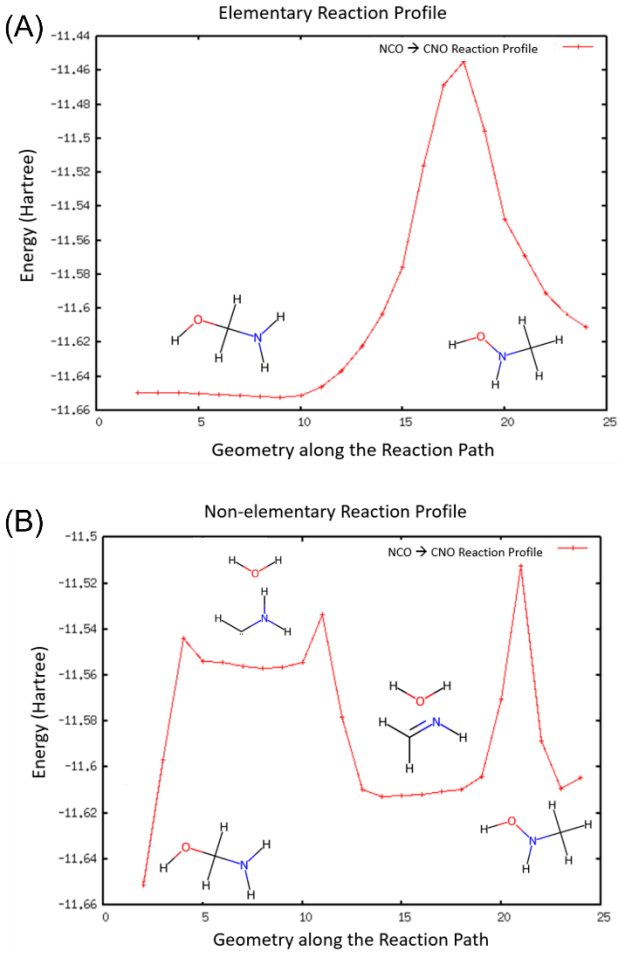

Figure 5. (A), energy profile for the elementary reaction NCO → CNO; (B), energy profile for a pathway leading from NCO to CNO with multiple elementary steps.

Subfigure (A) in **Figure 5** shows an energy profile for which the reaction occurs in one elementary step, with an activation energy of about 0.20 E_h. Subfigure (B) in Figure 5 shows a different energy profile, in which the overall transformations take place in a non-elementary way. Two intermediates connect reactant and product: NCO → N[CH].O → C=N.O → CNO. Intermediate C=N.O also exists in the generated network while intermediate N[CH].O does not appear because this aminocarbene species cannot be formed with a "two-bonds-breaking-two-bonds-forming" transformation since it does not fulfill the octet rule. The activation energy needed to make the non-elementary reaction happen is about 0.14 E_h.

Figure 5, therefore, shows that the path with the lowest number of intermediates or nodes does not necessarily coincide with the minimum energy path. This is why we do not recommend extracting the shortest paths by using the Dijkstra algorithm[55] when the intention is to calculate activation energies. Kim et al.[42] have previously suggested to combining Dijkstra's algorithm with

Yen's algorithm.[63] This would allow considering reaction paths with a larger number of nodes in case the minimum energy path does not correspond with a path with the minimum number of nodes. Nonetheless, such information cannot be known a priori. Instead, quantum-chemical calculations must be applied to elucidate the correspondence between the number of nodes and the minimum energy path. In the last instance, this means that paths with a larger number of nodes should be explored from the very beginning with expensive quantum calculations to increase the probability of finding the minimum energy path. We conclude that although BFS and Yen's algorithm can both overestimate and underestimate the reaction network, they are still less aggressive than Dijkstra's algorithm when trying to keep the minimum energy path with an unknown number of nodes in the network, and that the problem of reduction of the chemical network cannot be decoupled from the process of calculation of the activation energy using quantum chemistry.

Figures of reaction networks are located in SI section.

Network reduction via functional groups detection:

For system 5, we have also provided a shorter version of the database, which has been reduced to the minimum expression that allows the product formation from the reactant. The complete database, as well as its reduced version, are located in SI. **Table 2** shows the number of nodes and CPU time obtained when working with the complete database and its adapted reduced version for system 5.

Table 2. Network size and CPU time derived from system 5 when applying the complete database of functional groups **and a reduced version of** it **for systems 5**.

Database	Network Size	Reduced network size with BFS	CPU time (hh:mm:ss)
Complete	1185	189	02:57:50
Reduced	1033	103	02:39:56

By looking at the network size, we can see that the total number of nodes in the network decreases from 1185 to 1033. We also see that the reduced network size with BFS decreases from 189 to 103. We can also see a decrease in computational time from 02:57:50 to 02:39:56. We have been able to reduce the network size by 45%, which can make the posterior transition state search more feasible. This alternative however requires user knowledge about the interesting reactivity of the system under study, and although we can now consider paths beyond the PMCD we can miss important ones by limiting reactivity.

To conclude with the application of graph theory to network generation and network reduction, we can ensure that our method will provide a reaction network that does not miss any possible connectivity (at the moment with the caveat that only neutral species that fulfil the octet rule are located) while reducing computational time. Network reduction procedures such as functional group databases, BFS, or the Dijkstra algorithm may affect the results, so it is important to lead future movements to the development of new reduction strategies.

Transition State calculations with Quantum Chemistry

Table 3 shows the percentage of success of finding transition states for the different reactions in the different chemical systems. When the GSM method suggested that a given step was not an elementary reaction, the transition state search was not carried out, and therefore for these steps, the algorithm was not considered as having failed. Transition states for system 4 were calculated in the reduced network generated with BFS which contains 32 reactions, and not for the three reactions shown in the picture after applying Dijkstra algorithm.

Table 3. Method performance for the 5 systems. The average percentage of success was 91%. By executing Pysisyphus GSM calculation with a level of theory of DFT-B3LYP[59] and the basis set 6-31G(d, p) for those systems that failed, the percentage of success increases to 94%.

System	1	2	3	4	5
Number of Reactions	5	10	12	54	3
TS found	3	5	4	16	3
TS not found	0	0	1	2	0
Non-elementary Reaction	2	5	7	36	0

Percentage of Success	100%	100%	80%	89%	100%

We first see that only a small fraction of reactions correspond to elementary reactions as a result of a large number of non-elementary reactions described by ionic or carbene-like intermediates (which are not included with the two-bonds-breaking-two-bonds forming transformation). Nonetheless, our method will always ensure that if a non-elementary reaction can be decomposed into elementary reactions of neutral intermediates, such intermediates will appear as nodes in the original network. Important intermediates will also remain after a reduction with the BFS algorithm if the input chemical distance parameter d_{input} and the functional group database have been properly set. For the cases that ionic or other non-octet species exist as intermediates of those alternative elementary multistep paths, a possible reaction path that connects reactant and product for that specific non-elementary reaction may not exist. This is why the user must be sure that our method is applied to reaction systems that take place under non-ionic formation conditions.

Table 4 shows the CPU times of the different steps performed along the different steps of our method.

Table 4. CPU times for the different steps.

System	Network Generation Time	Conformer Orientations Generation	GSM GFN2-xTB Median	GSM GFN2-xTB Highest Value	SubautoTS DFT-B3LYP Median	SubautoTS Highest Value	TOTAL
1	00:00:03	00:02:37	00:04:07	00:08:48	00:08:31	00:19:41	00:31:15
2	00:00:21	01:01:28	00:09:46	00:16:19	01:27:13	01:59:13	03:17:53
3	00:00:51	02:25:51	00:08:41	00:11:00	08:13:16	13:53:11	16:31:35
4	00:08:12	07:44:58	01:00:59	01:39:00	07:30:23	09:24:38	19:13:12

By looking at the times needed for Network Generation in **Table 4**, we can see the level of performance of our method. We also see that the generation of different conformer orientations for systems with a large number of atoms becomes the bottleneck at least within the steps prior to the search for transition states. This step involves the 3D molecular coordinate generation, a Butina's clusterization over hundreds or thousands of stabilized conformers with molecular mechanics, and a GFN2-xTB geometry optimization.

We proceed to analyze the results based on their classification as non-elementary (row 5 in **Table 3**: Non-elementary reaction) and elementary (rows 3 and 4 in **Table 3**: TS found and TS not found).

Non-elementary reaction classification:

Reactions that fall into the group of non-elementary reactions were those whose lowest activation energy value came from a non-elementary reaction path. Increasing the number of energy paths to be considered for a specific reaction (this input parameter was set to 6 in our calculations) allows a more accurate classification, as well as increases the success possibilities of finding a transition state. When a reaction is labeled as non-elementary, no further calculations took place since intermediates that connect reactant and product by elementary reactions are expected to be described in the reaction network.

42 out of 50 non-elementary reactions were labeled as non-elementary because energy paths with more than one transition state proved to have a lower activation energy than those with a single transition state for that specific reaction, or because every reaction path contained more than one transition state. For the other 8 reactions, the method failed to find a suitable energy path that connects the reactant with the product when applying the GSM with the GFN2-xTB level of theory. For each of these 8 reactions, the same calculation was repeated for the 6 different reactant conformers, this time again using the GSM but with the DFT-B3LYP[59] level of theory. Results showed that none of the resultant energy profiles provided a valid transition state (there was more than one imaginary frequency or simply the calculation crashed). This can be due to several possibilities: lack of a reactant

or product stability, lack of a transition state, or lack of an energy path that connects the reactant and the product at those levels of theory. In these scenarios, increasing the level of theory or the number of initial relative orientations of the reactant with respect to the product could bring better results. We nonetheless labeled those scenarios as non-elementary since the highest level of theory we considered (DFT-B3LYP) did not find a valid reaction path that connects reactant and product by an elementary process.

Elementary reaction classification:

Our method found 31 transition states out of the 34 reactions classified as elementary. Every transition state was successfully validated with an intrinsic reaction coordinate that linked it to the expected reactant and product structures. When the GSM at the GNF2-xTB level with posterior refinement with *SubautoTS* at the DFT (B3LYP) level of theory failed, the GSM reaction path was recalculated with the *Pysisyphus* software,[64] which can apply the GSM algorithm with a range of different levels of theory – here we used the same B3LYP-DFT approach as used above. We could see that for one out of the three systems that failed, Pysisyphus was able to directly find the transition state at the highest level of theory that we considered (DFT -B3LYP). For such a calculation, we reuse the input orientations of reactant and product that showed the best performance when finding the lowest energy barrier with the GSM with the lower GFN2-xTB level of theory. This way, we take advantage of previous results to increase the probability of rapidly finding a transition state at the higher level of theory. Regarding the two remaining failed calculations (Cc1ccccc1NC(=O)C → Cc1ccccc1N1OC1Cl and CC(=O)O.CCCNC(C)=O → CC(=O)O.CCCC1(C)NO1 from systems 3 and 4 respectively), both of them include as reactant or product a species containing a three-membered ring NCO, which can be difficult to describe with our chosen levels of theory. Again, higher levels of theory will allow checking if the corresponding reaction path actually exists, or if it turns out to be non-elementary by allowing more reaction paths to be explored.

There were also transition state calculations that were successful when using our transition state searching procedure (GMS_GFN2-xTB + SubautoTS_DFT-B3LYP) and that failed when directly applying GSM with the higher level of theory DFT-B3LYP as implemented in Pysisyphus. Such results, as well as computational times, are shown in the SI, where we can see that our strategy of mixing levels of theory by (1) obtaining a suitable elementary reaction path with the lowest activation energy with the growing string method at a low level of theory, and (2) refining the transition state with *SubautoTS* with the higher *DFT-B3LYP* level of theory outperforms in computational time the growing string method when it is directly applied at the higher level of theory *DFT-B3LYP* as it is implemented in Pysisyphus. We also show that *SubautoTS* found the correct transition state in every single scenario except in three. From those, Pysisyphus found the correct TS structure in system 4 - elementary reaction r40; on the other hand, *DFT-B3LYP* as it is implemented in Pysisyphus found problems when trying to describe 14 out of 29 elementary reactions that were well described by our method. In the third column (GSM GFN2-xTB + SubautoTS DFT-B3LYP) we consider *GSM GFN2-xTB* computational time, while we have not added this time to the fourth column (Pysisyphus GSM DFT-B3LYP) although we believe it benefits from the input that is provided. Nonetheless, it is also possible that a valid molecular orientation in the *GFN2-xTB* level of theory harms results when working at the *GSM DFT-B3LYP* level of theory.

Conclusions

The novel method proposed in this paper is able to automatically generate reaction networks, in which different chemical species are connected according to elementary reactions by using graph theory. Quantum-chemical calculations are then used to locate TSs for every pair of connected nodes in the network. This results in the chemical exploration of potential energy surfaces. The main aspects to be highlighted are:

1) During the network generation phase, by directly working in the bond order space (which fully maps the chemical space) we obtain faster chemical exploration than in previous work.[14,21,42] We have also implemented a database with functional groups that guides the chemical exploration to selectively extract relevant paths based on reactivity, with the correspondent reduction in the time of exploration. A second database allows the obviation of paths that contain non-desired molecules based e.g. on toxicity or known thermodynamic inaccessibility under the temperature/pressure conditions, thereby avoiding slowing down the calculations for these species.

2) During the transition state search phase, a quantum-chemical calculation pipeline is used to efficiently find reliable transitions states at high levels of theory in an unsupervised manner. This is done by starting from a lower level of theory during the expensive phase of reaction path optimization with the growing string method, and later refining the structure and energy at a higher level of theory.

We have tested the performance of our method by studying chemical systems of different sizes regarding the number of atoms and elementary reactions from reactant to product. For the systems under study (up to 26 atoms and 3 elementary reactions connecting reactant with product), the presented method shows success of 91.70%. Although this success rate is high for the automation of such a complicated task, and reaction networks with hundreds of transitions states can now be calculated in our computational center in less than a week, the size of the chemical space generated as a function of the system under study illustrates the main challenges: we have shown that small variations in the number of elementary reactions can have important repercussions regarding the large number of reactions to appear in the network. We have also shown that the reduction of large chemical networks becomes mandatory to avoid unnecessary and expensive quantum calculations. Current network reduction methods can nonetheless lead to the removal of energetically favorable reaction paths, and we believe that future work should adopt reduction techniques that rely on energetic information derived from fast quantum calculations. These limitations encouraged us to develop the present method, which speeds the computation of network generation and quantum calculations while ensuring high quality and reliable transition state results. At the same time, our method proves to be flexible by allowing users to customize several easily-understood global parameters. We also believe, that with the growth in computational power to be expected in the next years, transition state calculations will become faster to perform. This will enable us to extend our method so that it can also identify important ionic species. Although our method has been fully-automated once the calculation is sent, some knowledge such as interesting functional groups to be considered may be expected for its optimal functioning. We expect future versions to also automatically identify relevant functional groups.

Computational details

Regarding the reaction network generation, all simulations used a python-based code in combination with calls to C++ functions from the RDKit cheminformatics library.[56] Network parameters were set so that the growth of the network happens with the "layered growth" functionality; active atoms were detected according to the database shown in **Table S1** in SI for calculations 1,2,3,5. For the second calculation of system 5, the reduced database shown in **Table S2** in the SI was used instead; nodes in the reaction networks that contained non-desired molecules listed in **Table S3** in SI were neglected in every system; a reduction of the reaction network with the BFS algorithm was used in all five systems; a second reduction of the network with the Dijkstra algorithm was performed in systems 4 and 5; the number of orientations in reactant and product for every reaction was limited to 6.

Regarding the quantum calculations:

For the GSM calculation we used the tight-binding GFN2-xTB method developed by Grimme[57] with D4-ATM dispersion correction terms for the lower level of theory calculations; and the DFT B3LYP functional with the 6-31G(d,p) basis set and the D4 dispersion correction terms for the higher level of theory calculations, implemented in Pysisyphus and Turbomole. [64,65] The GSM setup was always double-ended, with a path containing 25 nodes including endpoints when using GFN2-xTB and 18 nodes when using DFT-B3LYP, which grows from both endpoints with a node spacing of 5.0. A maximum of 200 iterations was allowed, with a maximum number of 30 optimization steps per growth step in GFN2-xTB, and 20 optimization steps per growth step in DFT B3LYP. The RMS force criteria on the TS node for string convergence was set to 0.0005 $E_h\ a_0^{-1}$ with a permitted maximum value of the perpendicular gradient set to 0.1 $E_h\ a_0^{-1}$ for the addition of new nodes. Finally, reduction of the optimization step was not allowed; having intermediate detection values higher than 2.0 kcal mol^{-1} while bond-breaking was allowed.

The geometry optimization of transition state structures obtained from the GSM with the GFN-xTB2 took place by applying the DFT-B3LYP functional with the 6-31G(d,p) basis set and the D4 dispersion correction terms with Turbomole.[65]

Refinement of transition states took place by also applying the DFT-B3LYP functional with the 6-31G(d,p) basis set implemented in Turbomole[65] and the Grimme dispersion correction terms D3 with Becke-Jones (BJ) damping, together with an in-house optimization algorithm based on Monte Carlo optimization. The frozen bonds in the initial optimization were the four active bonds involved in the "two-bonds-breaking-two-bonds-forming" transformation, which are the ones later distorted in the Monte Carlo transition state search.

A comparison of computational time between our method and the Pysisyphus setup is shown in **Table 4** in SI.

All simulations were performed in Covestro's computer cluster installations. The CPUs are Intel Xeon Gold at 3.3 GHz. All used nodes have 96 GB of memory. No parallelization was performed.

Conflicts of interest

The authors declare no competing financial interest.

References

[1] J. C. Lorquet, *J. Chem. Phys.*, **2019**, 150(16), DOI:10.1063/1.5092859.

[2] K. Fukui, *J. Phys. Chem.*, **1970**, 74(23), 4161, DOI:10.1021/j100717a029.

[3] R. B. Best, G. Hummer, *Proc. Natl. Acad. Sci. U. S. A.*, **2005**, 102(19), DOI:10.1073/pnas.0408098102.

[4] A. Behn, P. M. Zimmerman, A. T. Bell, M. Head-Gordon, *J. Chem. Phys.*, **2011**, 135(22), DOI:10.1063/1.3664901.

[5] G. Henkelman, B. P. Uberuaga, H. Jónsson, *J. Chem. Phys.*, **2000**, 113(22), DOI:10.1063/1.1329672.

[6] T. Asada, N. Sawada, M. Haruta, S. Koseki, *Chem. Phys. Lett.*, **2021**, 775, DOI:10.1016/j.cplett.2021.138658.

[7] R. Van de Vijver, J. Zádor, *Comput. Phys. Commun.*, **2020**, 248, DOI:10.1016/j.cpc.2019.106947.

[8] S. Maeda, Y. Harabuchi, Y. Ono, T. Taketsugu, K. Morokuma, *Int. J. Quantum Chem.*, **2015**, 115(5), DOI:10.1002/qua.24757.

[9] R. J. Hall, P. N. Mortenson, C. W. Murray, *Prog. Biophys. Mol. Biol.*, **2014**, 116(2–3), DOI:10.1016/j.pbiomolbio.2014.09.007.

[10] C. W. Coley, R. Barzilay, T. S. Jaakkola, W. H. Green, K. F. Jensen, *ACS Cent. Sci.*, **2017**, 3(5), DOI:10.1021/acscentsci.7b00064.

[11] S. Stocker, G. Csányi, K. Reuter, J. T. Margraf, *Nat. Commun.*, **2020**, 11(1), DOI:10.1038/s41467-020-19267-x.

[12] G. N. Simm, A. C. Vaucher, M. Reiher, *J. Phys. Chem. A*, **2019**, 123(2), DOI:10.1021/acs.jpca.8b10007.

[13] P. L. Kang, Z. P. Liu, *iScience*, **2021**, 24(1), DOI:10.1016/j.isci.2020.102013.

[14] Y. Kim, J. W. Kim, Z. Kim, W. Y. Kim, *Chem. Sci.*, **2018**, 9(4), DOI:10.1039/c7sc03628k.

[15] S. Habershon, *J. Chem. Theory Comput.*, **2016**, 12(4), 1786, DOI:10.1021/acs.jctc.6b00005.

[16] C. W. Coley, R. Barzilay, T. S. Jaakkola, W. H. Green, K. F. Jensen, *ACS Cent. Sci.*, **2017**, 3(5), 434, DOI:10.1021/acscentsci.7b00064.

[17] D. S. De, M. Krummenacher, B. Schaefer, S. Goedecker, *Phys. Rev. Lett.*, **2019**, 123(20), DOI:10.1103/PhysRevLett.123.206102.

[18] Z. Mihalić, D. Veljan, D. Amić, S. Nikolić, D. Plavšić, N. Trinajstić, *J. Math. Chem.*, **1992**, 11(1), DOI:10.1007/BF01164206.

[19] K. T. Schütt, H. E. Sauceda, P. J. Kindermans, A. Tkatchenko, K. R. Müller, *J. Chem. Phys.*, **2018**, 148(24), DOI:10.1063/1.5019779.

[20] P. Schwaller, A. C. Vaucher, T. Laino, J. L. Reymond, *Mach. Learn. Sci. Technol.*, **2021**, 2(1), DOI:10.1088/2632-2153/abc81d.

[21] I. Ismail, H. B. V. A. Stuttaford-Fowler, C. Ochan Ashok, C. Robertson, S. Habershon, *J. Phys. Chem. A*, **2019**, 123(15), DOI:10.1021/acs.jpca.9b01014.

[22] Y. Kim, W. Y. Kim, *Bull. Korean Chem. Soc.*, **2015**, 36(7), DOI:10.1002/bkcs.10334.

[23] Y. Kim, S. Choi, W. Y. Kim, *J. Chem. Theory Comput.*, **2014**, 10(6), DOI:10.1021/ct500136x.

[24] K. L. M. Drew, H. Baiman, P. Khwaounjoo, B. Yu, J. Reynisson, *J. Pharm. Pharmacol.*, **2012**, 64(4), DOI:10.1111/j.2042-7158.2011.01424.x.

[25] P. G. Polishchuk, T. I. Madzhidov, A. Varnek, *J. Comput. Aided. Mol. Des.*, **2013**, 27(8), DOI:10.1007/s10822-013-9672-4.

[26] E. J. Llanos, W. Leal, D. H. Luu, J. Jost, P. F. Stadler, G. Restrepo, *Proc. Natl. Acad. Sci. U. S. A.*, **2019**, 116(26), DOI:10.1073/pnas.1816039116.

[27] M. Liu, A. Grinberg Dana, M. S. Johnson, M. J. Goldman, A. Jocher, A. Mark Payne, C. A. Grambow, K. Han, N. W. Yee, E. J. Mazeau, K. Blondal, R. H. West, C. Franklin Goldsmith, W. H. Green, *J. Chem. Inf. Model.*, **2021**, 61(6), 2686, DOI:10.1021/acs.jcim.0c01480.

[28] A. L. Dewyer, A. J. Argüelles, P. M. Zimmerman, Methods for exploring reaction space in molecular systems, *Wiley Interdisciplinary Reviews: Computational Molecular Science*, *8*. 2018.

[29] S. Maeda, K. Ohno, K. Morokuma, *J. Chem. Theory Comput.*, **2009**, 5(10), DOI:10.1021/ct9003383.

[30] P. L. Bhoorasingh, R. H. West, *Phys. Chem. Chem. Phys.*, **2015**, 17(48), DOI:10.1039/c5cp04706d.

[31] C. F. Goldsmith, R. H. West, *J. Phys. Chem. C*, **2017**, 121(18), DOI:10.1021/acs.jpcc.7b02133.

[32] R. L. Hilderbrandt, *Comput. Chem.*, **1977**, 1(3), DOI:10.1016/0097-8485(77)85008-0.

[33] S. Kale, O. Sode, J. Weare, A. R. Dinner, *J. Chem. Theory Comput.*, **2014**, 10(12), DOI:10.1021/ct500852y.

[34] C. W. Gao, J. W. Allen, W. H. Green, R. H. West, *Comput. Phys. Commun.*, **2016**, 203, DOI:10.1016/j.cpc.2016.02.013.

[35] Y. V. Suleimanov, W. H. Green, *J. Chem. Theory Comput.*, **2015**, 11(9), 4248, DOI:10.1021/acs.jctc.5b00407.

[36] S. Maeda, T. Taketsugu, K. Morokuma, *J. Comput. Chem.*, **2014**, 35(2), DOI:10.1002/jcc.23481.

[37] Y. Guan, V. M. Ingman, B. J. Rooks, S. E. Wheeler, *J. Chem. Theory Comput.*, **2018**, 14(10), DOI:10.1021/acs.jctc.8b00578.

[38] R. A. Jara-Toro, G. A. Pino, D. R. Glowacki, R. J. Shannon, E. Martínez-Núñez, *ChemSystemsChem*, **2020**, 2(1), DOI:10.1002/syst.201900024.

[39] K. Ohno, S. Maeda, *Phys. Scr.*, **2008**, 78(5), DOI:10.1088/0031-8949/78/05/058122.

[40] B. Peters, A. Heyden, A. T. Bell, A. Chakraborty, *J. Chem. Phys.*, **2004**, 120(17), DOI:10.1063/1.1691018.

[41] L. D. Jacobson, A. D. Bochevarov, M. A. Watson, T. F. Hughes, D. Rinaldo, S. Ehrlich, T. B. Steinbrecher, S. Vaitheeswaran, D. M. Philipp, M. D. Halls, R. A. Friesner, *J. Chem. Theory Comput.*, **2017**, 13(11), 5780, DOI:10.1021/acs.jctc.7b00764.

[42] J. W. Kim, Y. Kim, K. Y. Baek, K. Lee, W. Y. Kim, *J. Phys. Chem. A*, **2019**, 123(22), DOI:10.1021/acs.jpca.9b02161.

[43] P. M. Zimmerman, *J. Comput. Chem.*, **2013**, 34(16), DOI:10.1002/jcc.23271.

[44] A. L. Dewyer, A. J. Argüelles, P. M. Zimmerman, *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, **2018**, 8(2), DOI:10.1002/wcms.1354.

[45] P. M. Zimmerman, *J. Comput. Chem.*, **2015**, 36(9), DOI:10.1002/jcc.23833.

[46] P. Zimmerman, *J. Chem. Theory Comput.*, **2013**, 9(7), 3043, DOI:10.1021/ct400319w.

[47] L. V. A. Hale, T. Malakar, K.-N. T. Tseng, P. M. Zimmerman, A. Paul, N. K. Szymczak, *ACS Catal.*, **2016**, 6(8), 4799, DOI:10.1021/acscatal.6b01465.

[48] B. R. Ellington, B. Paul, D. Das, A. K. Vitek, P. M. Zimmerman, E. Neil G. Marsh, *ACS Catal.*, **2016**, 6(5), 3293, DOI:10.1021/acscatal.6b00592.

[49] J. R. Ludwig, S. Phan, C. C. McAtee, P. M. Zimmerman, J. J. Devery, C. S. Schindler, *J. Am. Chem. Soc.*, **2017**, 139(31), 10832, DOI:10.1021/jacs.7b05641.

[50] A. L. Dewyer, P. M. Zimmerman, *ACS Catal.*, **2017**, 7(8), DOI:10.1021/acscatal.7b01390.

[51] M. L. Smith, A. K. Leone, P. M. Zimmerman, A. J. McNeil, *ACS Macro Lett.*, **2016**, 5(12), 1411, DOI:10.1021/acsmacrolett.6b00886.

[52] E. Martínez-Núñez, *Phys. Chem. Chem. Phys.*, **2015**, 17(22), DOI:10.1039/c5cp02175h.

[53] L. P. Wang, A. Titov, R. McGibbon, F. Liu, V. S. Pande, T. J. Martínez, *Nat. Chem.*, **2014**, 6(12), DOI:10.1038/nchem.2099.

[54] P. Burkhardt, *ACM Trans. Knowl. Discov. Data*, **2021**, 15(5), DOI:10.1145/3446216.

[55] E. W. Dijkstra, *Numer. Math.*, **1959**, 1(1), DOI:10.1007/BF01386390.

[56] G. Landrum, RDKit: Open-Source Cheminformatics Software, *Http://Www.Rdkit.Org/*. 2021.

[57] C. Bannwarth, S. Ehlert, S. Grimme, *J. Chem. Theory Comput.*, **2019**, 15(3), 1652, DOI:10.1021/acs.jctc.8b01176.

[58] N. M. O'Boyle, C. Morley, G. R. Hutchison, *Chem. Cent. J.*, **2008**, 2(1), DOI:10.1186/1752-153X-2-5.

[59] T. Yanai, D. P. Tew, N. C. Handy, *Chem. Phys. Lett.*, **2004**, 393(1–3), DOI:10.1016/j.cplett.2004.06.011.

[60] S. Grimme, S. Ehrlich, L. Goerigk, *J. Comput. Chem.*, **2011**, 32(7), DOI:10.1002/jcc.21759.

[61] J. Gamez, M. Leven, WO2020079093, 2020.

[62] D. Weininger, *J. Chem. Inf. Comput. Sci.*, **2002**, 28(1), 31, DOI:10.1021/ci00057a005.

[63] J. Y. Yen, *Manage. Sci.*, **1971**, 17(11), DOI:10.1287/mnsc.17.11.712.

[64] J. Steinmetzer, S. Kupfer, S. Gräfe, *Int. J. Quantum Chem.*, **2021**, 121(3), DOI:10.1002/qua.26390.

[65] F. Furche, R. Ahlrichs, C. Hättig, W. Klopper, M. Sierka, F. Weigend, *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, **2014**, 4(2), DOI:10.1002/wcms.1162.

An automated method for graph-based chemical space exploration and transition state finding.

Pablo Ramos Sánchez,[1] Jeremy N. Harvey,[2] and Jose A. Gámez*[1]

Correspondence to: Jose A. Gámez (E-mail: *jose.gamez@covestro.com*)

[1] *Covestro Deutschland AG, Leverkusen, Germany*
[2] *Department of Chemistry, KU Leuven, Leuven, Belgium*

Graphical abstract

A strategy for automatically exploring chemical space is presented. The method efficiently combines graph theory and quantum-chemical techniques to reduce the required human expertise and computational time for finding minima and saddle points in a potential energy surface. This is done by applying graph elementary transformations over automatically detected functional groups in molecules. Transition states in the resulted reaction network are then automatically found at a low level of theory to be later more accurately described at a higher level.