

Special  
Collection

# Traversing Dense Networks of Elementary Chemical Reactions to Predict Minimum-Energy Reaction Mechanisms

Christopher Robertson,\* Idil Ismail, and Scott Habershon<sup>\*,[a]</sup>

Numerous different algorithms have been developed over the last few years which are capable of generating large, dense chemical reaction networks describing the inherent chemical reactivity of a collection of discrete molecules. For all elementary reactions in a given reaction network, reaction rate calculations, followed by direct micro-kinetic modelling, enables one to predict macroscopic outcomes (e.g. rate laws, product selectivity) based on atomistic input data. However, for chemical reaction networks containing thousands of reactant molecules, such simulations can be extremely time-consuming;

in addition, the complex coupled time-dependence of molecular concentrations can present challenges when seeking essential mechanistic features. In this Article, we instead present an algorithm which seeks to predict the “most likely” reaction mechanism, or competing mechanisms, connecting any two user-selected reactant and product species, given a previously-generated reaction network as input. The approach is successfully tested for reaction networks (containing tens of thousands of possible reactions) describing the carbon monoxide oxidation on platinum nanoparticles.

## 1. Introduction

Chemical reaction mechanisms, namely the sequence of elementary chemical reactions which connect sets of reactant and product molecules, occupy a central position in synthesis, catalysis and biochemistry.<sup>[1–3]</sup> Characterization of a reaction mechanism is often fundamental to technological step-changes; for example, better understanding of the chemical mechanisms associated with the emergence of antibacterial resistance in microbes can in principle be leveraged to develop next-generation antibiotics, while mechanistic insights can also help decipher, and ultimately improve, enantioselectivities of organometallic-complex-catalysed in organic transformations which might be used to synthesize these new medicines. As such, investigation of chemical reaction mechanisms remains a perennial challenge to both experimental and computational chemists.


Increasingly, computational chemistry tools are being developed which enable one to generate large networks of inter-related chemical reactions as a means of interrogating emergent mechanisms in complex chemical systems.<sup>[4–15]</sup> These so-called reaction-discovery tools generally fall into one of three broad (and often overlapping) categories. First, many *heuristic* (or rules-based) reaction-discovery tools have been suggested over the last couple of decades, most commonly based on the idea of representing molecules as graphs (or adjacency matrices) and chemical reactions as matrix operations acting on

those graphs. Examples of these approaches include the Reaction Mechanism Generator (RMG) developed by Green and co-workers,<sup>[8,16,17]</sup> the Zstruct approach by Zimmerman,<sup>[18–20]</sup> recent work by Reiher and co-workers,<sup>[11,21,22]</sup> the basin-hopping Monte Carlo strategy adopted by Kim and co-workers,<sup>[23,24]</sup> and our own recent work on graph-driven sampling schemes to investigate both open-ended and double-ended reaction networks.<sup>[6,9,14]</sup> This non-exhaustive list of examples, which spans approaches ranging from Hamiltonian-based sampling schemes to brute-force reaction-network generation, demonstrates the flexibility of graph-based descriptors in constructing algorithms for investigating reaction networks.

The second broad class of reaction-discovery tools are those based on artificial intelligence (AI) tools, most commonly deep learning neural network strategies. In particular, the last few years have seen the emergence of deep learning tools which can learn to predict the outcomes of (typically organic) reactions given experimental data such as reaction yields, often available in large chemical databases or electronic notebooks.<sup>[25–28]</sup> Once trained on existing experimental data, these artificial neural networks can subsequently be used to, for example, predict synthetic routes to complex organic molecules which are often competitive with those generated by human experts. While challenges remain, such as the prediction of stereochemical outcomes, these tools are a prime example of computational research complimenting experiment design, and are surely set to become increasingly powerful in the near-future.

Finally, there are also several reaction-discovery methods which can be viewed as being physics-based in the sense that exploration of chemical space is driven by molecular dynamics (MD) or similar configurational sampling. A prominent example in this category is the transition-state searching with chemical dynamics simulations (TSSCDS) approach developed by Marti-

[a] Dr. C. Robertson, I. Ismail, Dr. S. Habershon  
Department of Chemistry and Centre for Scientific Computing, University of Warwick, Coventry, CV4 7AL, United Kingdom  
E-mail: C.Robertson@warwick.ac.uk  
S.Habershon@warwick.ac.uk

 An invited contribution to a Special Collection on the Computational Chemistry of Complex Systems

nez-Núñez, which integrates high-temperature MD simulations with graph-based post-processing to explore transition-states for chemical reactions, enabling construction of complex reaction networks such as those found in organometallic catalysis.<sup>[4,29,30]</sup> A second example in this category is the *ab initio* nano-reactor developed by Martinez and co-workers,<sup>[10]</sup> which uses periodic application of high-pressures during an *ab initio* MD simulations in order to drive generation of new chemical products; post-processing, again based on graph-based analysis, then allows construction of the reaction mechanisms which ultimately led to observed molecular product species.

Once generated (by any relevant method) chemical reaction networks can be supplemented with experimental or calculated reaction rate data for each elementary reaction step; from the computational viewpoint, this most commonly requires determination of the transition-state (TS) of each reaction, followed by application of transition-state theory (TST), using standard rigid-rotor/harmonic oscillator approximations, to evaluate rates.<sup>[3,31–33]</sup> The calculated (or experimentally-available) rates of each reaction in a chemical reaction network can then be used to perform microkinetic simulations, for example using the well-known Gillespie stochastic simulation algorithm.<sup>[34–36]</sup> Such calculations enable direct connection between microscopic rate constants and macroscopic outcomes; for example, one can monitor the time-dependent concentrations of the individual molecular species which constitute the reaction network, enabling prediction of rate laws, product selectivities, pressure dependence and temperature dependence.<sup>[37]</sup> Furthermore, analysis of reactive flux in such simulations enables one to readily draw conclusions about the emergent reaction mechanism connecting reactant and product molecules.<sup>[6]</sup>

However, it is at this stage that the computational burden associated with analysing large dense networks of chemical reactions becomes apparent. As we highlight below, chemical reaction networks, even those constructed for systems containing moderate numbers of reactive molecules, can very quickly grow to enormous sizes; this growth is simply a result of the combinatorial explosion in the number of possible chemical reaction possibilities that one can generate for a given collection of reactant molecules, as well as the associated conformational space of the reactant species. For example, reaction networks designed to model combustion and pyrolysis of mixed hydrocarbons can easily contain thousands of elementary chemical reactions, even if one limits interest to “chemically-relevant” reactions without considering wider exploration of more exotic reaction mechanisms.<sup>[16,38–41]</sup> This complexity is underlined below, where we show that more than  $10^4$  elementary reactions can be readily generated for carbon monoxide oxidation on small nano-particles.

This Article discusses a simple search strategy which can be used to extract the “most likely” competing reaction mechanisms leading to any given product from a set of input reactants, out of a large chemical reaction network. Here, initial reaction networks are first constructed from the output of our graph-driven sampling (GDS) algorithm, as described below and reported previously. Using a reactive force-field, namely ReaxFF,<sup>[42–45]</sup> for computational efficiency (at the obvious

expense of some accuracy), we subsequently evaluate the reaction energies and TS barriers for all reactions in the network, enabling us to provide some directionality and weights to all network edges. Finally, we introduce a series of network analyses, based on depth-first search (DFS), which can be used to extract the most likely reaction mechanisms leading to formation of any user-defined product species, given the thermodynamic and kinetic data available in the full network.

As an aside, we note that our approach of evaluating *all* reaction barriers in the generated network will reach an impasse when attempted using more accurate electronic structure methods; however, in the context of the current work, the use of reactive force-fields instead of *ab initio* methods is not a particularly important distinction, and we note that this computationally-cheaper approach can nevertheless serve as a way of quickly focus down onto a smaller number of likely reaction mechanisms for more detailed later analysis. As an alternative strategy, we also consider the approximation of activation energy barriers using the Brønsted-Evans-Polanyi (BEP) relation,<sup>[46–48]</sup> namely  $\Delta E_i = \alpha \Delta U_i$ , where  $\Delta U_i$  is the energy change of the *i*th reaction. We show that using such a simple approximation, the graph-search algorithm presented here qualitatively explores similar regions of the full chemical reaction network as it does when using actual (ReaxFF-calculated) activation energies. Finally, we note that this approach of solely using reaction energetics has been successfully exploited previously, such as in the work by Green and coworkers.<sup>[8,16,17]</sup>

The remainder of this paper is organised as follows. In Section 2 we present our mechanistic generation (MechGen) algorithm, the notable steps being:

1. Definition of the reaction system, primarily initial reactant molecules, allowed reaction classes and atomic valence ranges;
2. Constructing reaction networks using the previously-reported graph-driven sampling (GDS) algorithms (noting that, of course, any relevant reaction-discovery tool could equally be used here);
3. Trimming the reaction networks based on local network characteristics tailored for the specific choice of initial reactants and final product;
4. Traversing the pruned network to identify efficient reaction mechanisms leading to user-defined product species;
5. Constructing *reaction-trees* representing efficient individual reaction pathways.

In Section 3 we present applications of this approach to the catalytic oxidation of carbon monoxide on platinum clusters ( $\text{Pt}_n$ ,  $n = 1, 5, 7$ ); such systems already exhibit a challenging network of reactions, and portend our interest in using these techniques to study realistic and substantially larger clusters in future work. Finally, Section 4 summarizes our results and discusses the utility and limitations, as well as future improvements of this algorithm.

## 2. Methodology

In this Section, we highlight challenges and solutions associated with analysis of large chemical reaction networks. First, we briefly discuss our approach to generating the set of possible molecular species and chemical reactions which might emerge from a given initial set of reactant species, and the construction of a chemical reaction network (CRN). Next, we discuss a simple pruning-and-searching algorithm for identifying reaction mechanisms connecting a user-defined product to reactant species. Finally, we summarize by presenting pseudo-code for our approach.

### 2.1. Objective of the Algorithm

If the rate constants for all elementary reactions in a CRN are known, one can construct a microkinetic model and obtain yields for the different molecules under different initial concentrations and thermodynamic conditions. However, using the full CRN directly will often be unnecessarily costly in such analyses and, considering the approximate nature of the typical calculated rate constants, the accrued error in such simulations could make the results unreliable.

As an alternative to expensive analysis of the entire CRN, one could instead restrict focus on searching for only those competing reaction pathways that generate a user-defined product species of interest. As a concrete example, in the application discussed later we are interested in identifying reactions which lead to formation of carbon dioxide given carbon monoxide, molecular oxygen and platinum nanoparticles as reactants. Detailed evaluation and simulation of the full CRN is not necessary if one is interested in a particular specified reaction; the algorithms discussed below seek to find the most efficient traversals through a given CRN from a pre-defined set of reactant species to a specific product species. Pre-empting our view of CRNs in the form of tree-graphs, we will refer to these preselected set of reactants as *leaf-reactants* and product as *root-product*. Figure 1 outlines our overall

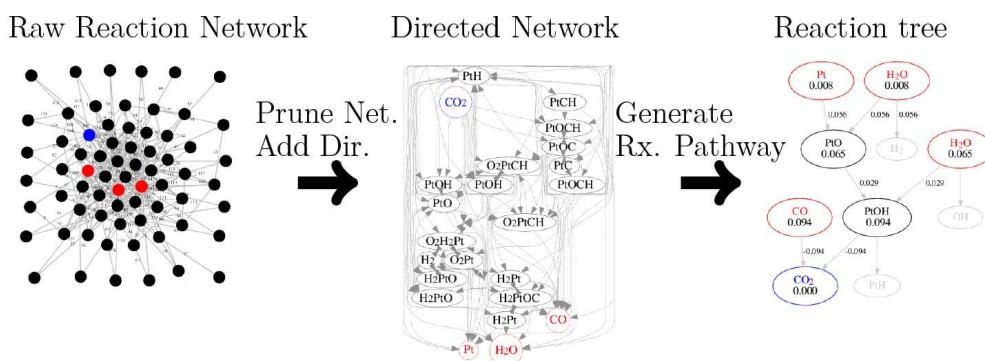
scheme to obtain competing reaction pathways from CRNs, as described below.

Our algorithm outputs a collection of the best competing reaction mechanism within the network that we shall call *reaction-trees* – a sequence of elementary reactions starting from the leaf-reactants to the root-product. Often the minimum energy reaction pathway, accounting for both thermodynamic and kinetic energy changes, is the ‘most likely’ mechanism for a reaction, and so it becomes imperative to find such minimum-energy mechanisms amongst the generated reaction-trees generated.

### 2.2. Initial Generation of Molecules and Reactions

The approach outlined below is a general scheme to try to extract likely reaction mechanisms leading to user-defined molecular species; exactly how the input CRN to be studied is initially generated is not relevant. Of course, underlying this study is the assumption that the initial CRN does indeed contain reaction-paths which connected targeted leaf-reactants and root-products.

In this Article, we use a modified version of the GDS scheme proposed previously;<sup>[6,14]</sup> the details of our previous graph-driven reaction exploration schemes have been given elsewhere, and we highlight here only the broad approach. In particular, rather than performing full Hamiltonian dynamics in order to construct reaction networks, as in our previous work, we instead use a much simpler scheme; given a particular reactant configuration and associated connectivity matrix (CM), we generate possible reaction CMs by performing chemically-allowed reaction operations on the reactant CM. As described in our recent work, initial molecular configurations of the new product species are generated using a graph-restraining potential (GRP), which simply imposes a given CM on a set of atomic Cartesian coordinates. Geometry optimization of the resulting reaction end-points then enables evaluation of the reaction energy change, and the new (product) CM and configuration is then used as the reactant CM for the next iteration. Duplicates are subsequently filtered out, and isomer-



**Figure 1.** Overview of our mechanism generation (MechGen) procedure. Following construction of a raw chemical reaction network (left), we prune and add directionality to the elementary reaction edge-tuples (as described in text) for a particular choice of leaf-reactants (in red) and root-product (blue) resulting in a Directed Network (middle). We finally calculate the most efficient reaction trees that produce the root-product from the leaf-reactants (right).

ization reactions are also added to the CRN for all GDS-generated molecules with identical CM. TS barriers are then evaluated for all reaction paths using the Auto-Nudge Elastic Band algorithm (Auto-NEB).<sup>[49,50]</sup>

### 2.3. Definition of CRNs

The CRNs defined here have vertices (or nodes) which represent unique molecular species; these species are energetically local minima, and we do not collect together geometric isomers of the same molecule (noting that, if desired, we could easily bunch together isomers in a more coarse-grained view). The edges in the CRNs represent either elementary (*i.e.* bond-forming or bond-breaking) reactions or any form of isomerization reactions (which do not involve changes to chemical bonding). Decisions about what constitutes a bond-breaking or bond-forming reaction are simply based on the geometric distance between the corresponding reactive atoms, as well as monitoring changes to the CMs of reactants and products.

Furthermore, we note that edges in our CRN must necessarily be treated as connected *tuples*; for example, in the case of the reaction  $A \rightarrow B + C$ , traversing along one of the edges of this reaction. For example, moving from vertex  $A$  to vertex  $B$  will, in the context of the algorithm described below, necessarily imply a simultaneous traversal along the second edge from vertex  $A$  to vertex  $C$ .

In what follows, to explore the different traversal routes possible within a CRN, we will use the DFS algorithm<sup>[51]</sup> as a way to systematically traverse the network, typically starting from the final root-product node.

### 2.4. Edge Weights of the CRN

As noted above, the overall aim of the approach discussed here is to traverse a large dense CRN in order to identify paths which are most likely to result in formation of a user-defined product species. As a result, it is essential that a measure of 'efficiency' can be associated with each elementary reaction and reaction mechanism in order to assess which reaction mechanisms might be worth investigating further using higher-level methods such as TST.

As a first approximation of overall mechanism efficiency here, we adopt the simplest approach of evaluating the effective first-order rate constants for every elementary reaction in the initially-generated CRN. This is, of course, an approximation; however, the effect of using a first-order approximation for second-order reactions will be to effectively make such reactions more efficient overall. If the aim is to simply provide a 'first-pass' screening of possible reaction mechanisms which run through the CRN towards defined products, this approximation seems sensible. Of course, improvements of this approximation can be incorporated into future work.

For a given reaction mechanism comprising  $N$  elementary reactions, we then define the 'efficiency' (or 'fitness') as the sum of lifetimes approximated from first-order rates,

$$\theta = \sum_{i=1}^N \tau_i, \quad (1)$$

where

$$\tau_i = \left[ e^{\frac{\Delta E_i}{k_B T}} \right]^{-1}. \quad (2)$$

Here,  $k_B$  is the Boltzmann constant and  $T$  is the temperature. So, a lower total lifetime  $\theta$  for a given reaction mechanism of  $N$  steps is identified as a more efficient mechanism.

As an aside, we note that the nature of our GDS scheme is such that we occasionally sample reaction steps which actually comprise more than one energy barrier. In such cases, for  $M$  effective activation barrier, we calculate the lifetime  $\tau_i$  for the combined step using,

$$\tau = \left[ \sum_i^M e^{\frac{\Delta E_i}{k_B T}} \right]^{-1}. \quad (3)$$

As noted previously, calculating the activation energy for a large ensemble of elementary reactions may not always be computationally affordable. In this Article, we will instead briefly explore an alternative method to screen for sensible reaction mechanisms before undergoing the expensive calculation of activation energies. In particular, we consider a single-parameter approximation based on the BEP relation. This relation suggests a proportionality between the activation energy for a reaction,  $\Delta E_i$ , and the reaction energy change  $\Delta U_i$ , namely

$$\Delta E_i = \alpha \Delta U_i,$$

where  $\alpha$  is some suitably chosen parameter. The utility of this relation is that the reaction energy change of each elementary reaction in a CRN is more straightforward to evaluate (requiring geometry optimization and energy evaluation) than activation barriers (additionally requiring TS-finding algorithms). Most importantly, we show below that the vertices and edges in the predicted reaction mechanisms extracted from a CRN using such an approximation broadly agree with those obtained with actual activation barriers. We emphasize here that our approach towards using the BEP relation is simply as a screening tool to remove candidate reaction-mechanisms which would have unfeasibly large reaction barriers; as such, we do not seek to optimize the value of  $\alpha$ , and we do not account for any possible constant shift factor for the barrier heights.

### 2.5. Adding Directionality and Pruning Edges in the CRN

Traversing the CRN using DFS by starting from the desired root-product, we expect that there will be many vertices that cannot lead to the desired leaf-reactants; such paths should be pruned to minimize the CRN and enable extraction of the unique set of mechanisms which lead to the target root-products. Similarly

one can also determine if an edge (*i.e.* elementary reaction) can lead to a product along none, one or both directions.

To account for this directionality in the network, and to prune irrelevant vertices and edges in the CRN, we adopt the following steps, as highlighted in Figure 2:

- **Activate vertices and add directionality to edges:** We perform a DFS starting from the desired product species, adding directional arrows to the edges as we traverse the network. Vertices which are not visited are pruned away (removed from the CRN).
- **Iteratively prune network:** Prune edge-tuples if they point to any vertices which: (i) have been marked as inactive, or (ii) are dead-ends which do not have any edges radiating outwards (unless it is an identified leaf-reactant).
- **Prune loops:** DFS traverse the network starting from every vertex, and stop when the vertex has more than one possible edge-tuple (reaction) available. If the DFS ends up in the initial vertex, prune the last edge.

The last two steps are repeated iteratively until no further vertex or edge pruning occurs, and it scales linearly with the size of the network. The above procedure ensures that the remaining CRN only contains directed edges that may, *via* some route, eventually lead to leaf-reactants; all other reaction pathways are removed.

## 2.6. Exploration of Pruned CRNs: Breadth-First Search and Depth-First Search

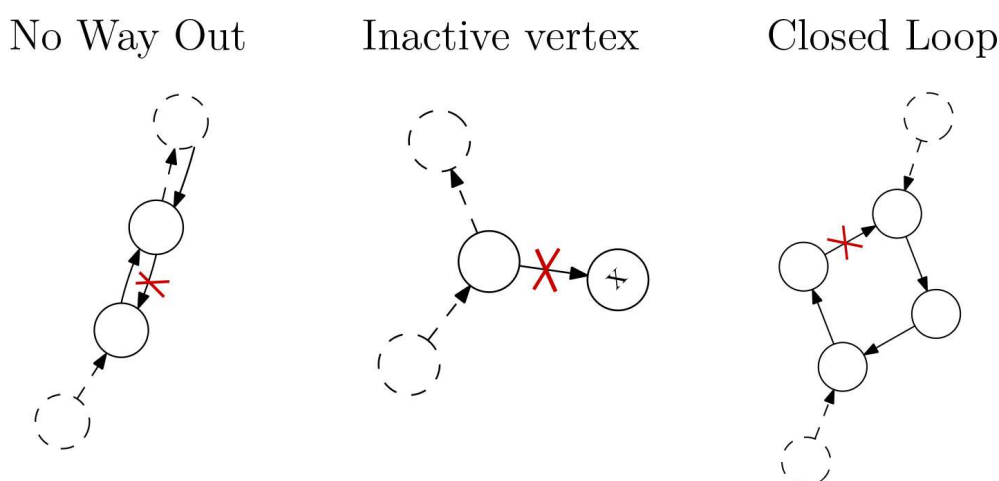
After pruning the initial CRN, the result is a reduced CRN which contains only the *essential* set of reactant molecules and reactions which participate in pathways connecting user-defined leaf-reactants and root-products. The next task is then to identify, and evaluate the efficiency of those reaction paths which definitively connect reactants and products.

Standard DFS or breadth-first search (BFS) algorithms are commonly used to ensure that one visits all vertices in a

connected network.<sup>[51]</sup> With DFS, one usually keeps track of all the vertices that have been visited in order to terminate the search along a given network pathway once a vertex is visited twice (*e.g.* as used in the previous subsection). The DFS search tree resulting from a DFS in a simple CRN, is shown in the middle cell of Figure 3.

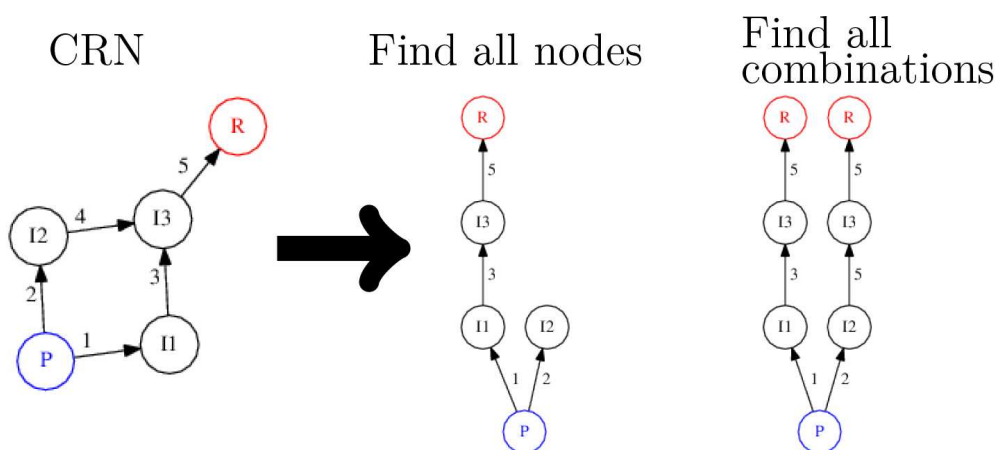
However, DFS/BFS can also be used to systematically explore *every possible trajectory* outwards from some initially-selected network vertex. Of course, if one allows for repetition of vertices visited, the possibilities are infinite; however, in the next subsection, we will discuss why only permitting a vertex (or edge) to appear once in each single-line-branch of the search tree is appropriate for our purposes. As a result of this modification (*i.e.* searching for every possible reaction mechanism moving outwards from some root-product), the DFS trees obtained differ from the ones where we keep track of all vertices visited, as shown in the rightmost cell of Figure 3. The combination of all possible routes between the root-product to a leaf-reactants scales at best exponentially with the size of the network; as a result, limiting the exploration depth of the DFS tree will be essential, as discussed below.

As an aside, we note that BFS is often used to find the shortest path between two vertices, which would suggest itself potentially ideal for the context of this work ('shortest' in this context relating to the efficiency). However, we are not simply interested in finding *only* the shortest (most efficient) path, but a collection of high-efficiency paths that represent potentially competing pathways. In other words, we want to explore many *possible combinations of short-lifetime edges* (reactions) leading from the root-product to leaf-reactants in the pruned CRN. BFS would require that we store all the possible edge-sequences as we traverse the CRN away from the root-product. However, this approach quickly becomes prohibitively expensive with every branch of the search tree which is explored, particularly with regards to disk storage. Furthermore, we wish the search algorithm to allow for the possibility of exploring long sequences of edges with weights corresponding to (potentially)



**Figure 2.** Three “dead-end” conditions under which edges of an initially-generated chemical reaction network can be pruned. The left-hand panel illustrates pruning of reverse reactions which lead nowhere, the middle panel represents pruning of vertices with no further active connections, and the right-hand panel illustrates pruning of closed loops.





**Figure 3.** The difference between the DFS search trees discussed in the text. The left-hand figure shows a simple chemical reaction network with a single indexed edge (reaction) coming out of each vertex. A DFS search tree (starting from the root-product in blue) that seeks to find all vertices will look like the reaction-tree in the centre of the Figure. The right-hand reaction-tree is that which would be produced if we seek to find all combinations of ways to reach the leaf-reactants *R* (in red) from the root-product.

short lifetimes  $\tau_i$ , which might reasonably correspond to realistic mechanisms; as a result, BFS is not ideal. In contrast, DFS has the advantage of not requiring one to store every possible traversal up to a given search-tree branch; this is the principal reason we have chosen to analyse our pruned CRNs using DFS, and is largely why we have opted to build our analysis algorithm around it.

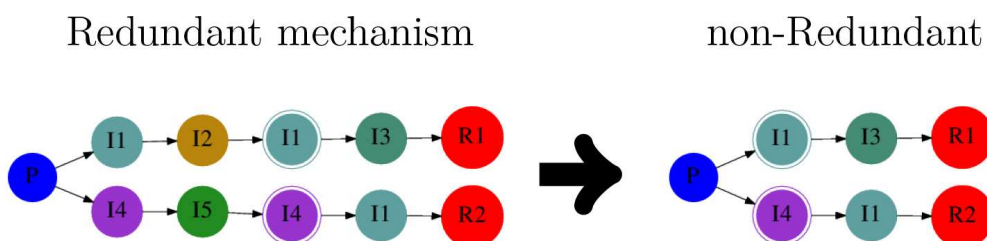
## 2.7. Constraining the Extent of DFS Exploration

When using DFS to generate distinct reaction pathways in a given pruned CRN, it is essential that each line sequence of visited vertices starting from the root-product and leading to the leaf-reactants of the DFS search tree does not repeat vertices. Even allowing for one repetition of an intermediate vertex implies a redundancy in the resulting proposed pathway; if an intermediate molecule appears twice in a direct sequence of reactions (*i.e.* the vertices taking us from current tip vertex of the DFS tree down to the root-product vertex), it means that there exists a more efficient mechanism which skips the entire

sequence of reactions between the first appearance of a molecule and its second (repeat) appearance. Figure 4 shows an example of a redundant mechanism on the left cell and a more sensible minimal-mechanism on the right. A similar consideration can be made by not allowing the same edge (reaction) to appear twice in the same complete branch of the DFS tree. However, it is permissible for the same vertices or edges (*i.e.* reactions) to appear in different branches of a DFS tree, as shown in the right cell of Figure 4.

Because of the exponential growth of reaction pathways possible in a large CRN (even a pruned one), we define four parameters that we use to constrain the DFS, as follows:

- For most reaction-mechanisms problems of interest, it is clear that there will exist some upper-bound for the total number of reaction steps which could sensibly comprise an overall mechanism. So, we set a threshold for the highest possible tree generation,  $G^{max}$ , during DFS.
- The energetic barriers along a given reaction sequence will also play an important role in guiding us towards sensible reaction mechanisms; reaction pathways with extremely long lifetimes (*i.e.* large effective activation energies) should be



**Figure 4.** The left-hand Figure shows a redundant mechanism allowed by repetition of vertices in the DFS search tree; the first reaction starting from the product (in blue) corresponds to a two-edge-tuple leading to intermediates *I1* and *I4*. Note that these intermediates appear again down each branch towards leaf-reactants *R1* and *R2* (both in red), respectively. The reappearance is highlighted as a second outer circle. Note that visiting vertices *I2* and *I5* on the left-hand mechanism is redundant, since a more efficient sequence is given by the right-hand mechanism. Note also that the right-hand mechanism permits vertex *I1* to appear twice, but along a different sequence of reactions.

viewed as unfavourable. Consequently, when the sum of lifetimes (Eq. 1) exceeds a threshold value  $\tau^{\max}$ , we truncate the search.

- We adopt the viewpoint that chemical reactions involving bond-breaking or bond-forming will, in general, have larger associated barriers than typical isomerization reactions which do not involve bond changes. Thus, we group together reactions and species describing isomerizations that do not involve chemical reactions. Consequently, there will be many branches of the DFS tree with many combinations of reaction sequences with short-lifetimes which only interconnect molecular isomers within an isomer-set. To counter this, we define  $I^{\max}$ , a threshold for the number of molecules that a branch in the DFS tree can contain from each isomer-set.
- Because  $G^{\max}$ ,  $I^{\max}$  and  $\tau^{\max}$  constrain the DFS 'on-the-fly', it is not possible to *a priori* determine how much exploration space these parameters will permit during DFS. The CPU-cost is ultimately largely dependent on this exploration space, and it is roughly proportional to the number of vertices visited during DFS. We therefore need a way of relating the parameters  $G^{\max}$ ,  $I^{\max}$  and  $\tau^{\max}$  to the 'cost' of the search. Here, we use the maximum number of vertices visited,  $V_{\text{vis}}^{\max}$ , as a way of constraining the extent of the search; this value is expected to be roughly proportional to overall computational expense. The user will typically be able to estimate  $V_{\text{vis}}^{\max}$  (knowing CPU resources), as well as  $G^{\max}$  and  $I^{\max}$  (putting some sensible limit on the nature of the mechanism sought). Assuming user determined  $G^{\max}$  and  $I^{\max}$ ,  $V_{\text{vis}}^{\max}$  will grow exponentially as a function of  $\tau^{\max}$ , although the latter parameter is not so easily estimated. We therefore calculate  $V_{\text{vis}}^{\max}(\tau^{\max})$  for a small number of values  $\tau^{\max}$  (without constructing reaction pathways, as discussed in the next subsection) and fit an exponential parameter in order to estimate a reasonable value of  $\tau^{\max}$  for the user provided  $V_{\text{vis}}^{\max}$ .

## 2.8. Generating Reaction-Tree Diagrams

The last section dealt with the exponential *cost of exploring* many possible combinations of edges towards leaf-reactants; here we shall address the challenge of *combining* edges together to construct complete reaction pathways.

Because edges come in tuples, meaning that an elementary reaction has often more than one product, the possible sequence of reactions leading from root-product to leaf-reactants are not simply a linear sequence of intermediates, but form a tree structure, branching out with every product of every reaction. We thus need to identify minimal reaction-trees within the pruned CRN which connect the root-product to sets of leaf-reactants. In such trees, each vertex starting from the root-product will have one and only one reaction (edge-tuple) directed towards leaf-reactants.

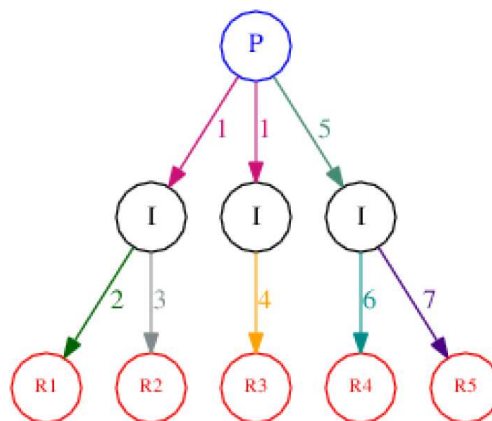
During DFS, for every vertex (*i.e.* intermediate molecule) there are a number of possible edge-tuples (reactions) available for traversal. In turn, every possible child vertex generated by a given choice of edge-tuple, will itself have edge-tuples available to it. In other words, every child vertex will generate all

combinations of reaction trees with themselves as 'roots', albeit one generation shorter; we refer to these shorter reaction-trees as *branches*.

Starting from some 'parent' vertex  $i$  (an intermediate or otherwise), *any one specific* edge-tuple  $R_{ij}$  will have  $n_e^{R_{ij}}$  edges (leading to child vertices). The number of possible reaction-trees that we can obtain from this edge-tuple  $R_{ij}$  is given by *every possible combination* from the  $B_i$  branches that each of the

$n_e^{R_{ij}}$  child vertices provides, that is  $B^{R_{ij}} = \prod_i B_i$ . The total number of reaction trees and branches available from such a parent vertex is the *sum* of all those combined branches from each of the  $n_e^i$  reactions available to this vertex, namely  $B_i = \sum_j n_j^{R_i} B^{R_j}$ . This evaluation of possible reactions is illustrated in Figure 5, which depicts a simple directed network (as might be generated from pruning of the raw CRN as described above) and shows four possible reaction trees which might be generated from it.

Our algorithm only begins to combine branches once the DFS has successfully reached a leaf-reactant. Each vertex may have multiple branches that lead to leaf-reactants, and these need to be combined in every possible way and returned to the parent vertex down the DFS tree. This is a source of combinatorial scaling which exacerbates an already-costly problem and we deal with it by choosing a maximum set of branches that every child vertex is permitted to give to its parent vertex,  $B^{\max}$ . Of this set of branches, those with best 'fitness' (that is, with lowest sum of lifetimes  $\sum_i \tau_i$ ) are chosen to be returned down the search tree. However, what constitutes the 'fitness' of a path is an approximate measure and it will not always be the case that the fittest path is the most realistic (*e.g.*



**Figure 5.** Example of a simple directed network that has a tree structure starting from the root-product (in blue and labelled with  $P$ ), with different reactions (edge-tuples, indexed and coloured) available to each vertex. The reaction-leaves are indexed in red and labelled with  $R$ . Every possible reaction-tree can be formed by choosing one edge-tuple from each vertex starting from the root-product. The indices of the reactions are chosen to describe how one would encounter them during a DFS search of this reaction-tree. There are four possible reaction-trees in this example. The leaf-reactants obtained from each possible reaction-tree available in this network are ( $R_1$ ,  $R_3$ ), ( $R_2$ ,  $R_3$ ), ( $R_4$ ) and ( $R_5$ ). In other words, reactions involving these species would ultimately lead to production of the product  $P$ .

termolecular reactions were found during the tests shown in section 3). Consequently, as well as passing the  $B^{max}(1 - \alpha^c)$  best branches, we also group the branches into different 'types' (described shortly) and pass an assortment of  $B^{max}\alpha^c$  branches of the best from each group (where  $\alpha^c \leq 1$  is some user-defined parameter).

## 2.9. Grouping Reaction Branches

Here, we present a simple scheme that we have used to *tree group* the collection of reaction branches so as to retain a 'greater diversity' of reactive pathways as the algorithm explores the pruned CRN, as well as to aid classification.

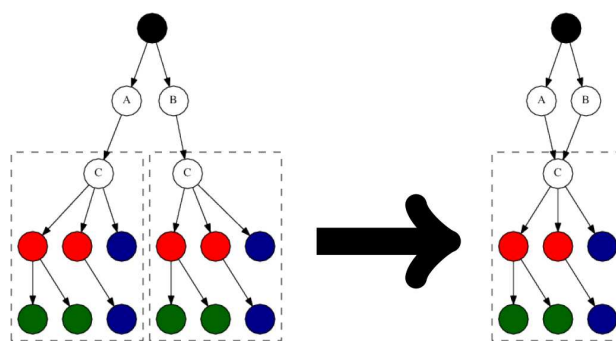
Let  $\vec{v}_i$  be a vector descriptor of some particular branch  $i$ , whose elements map to a vertex in the network, and whose entries give the relative energy difference between the root-product and that intermediate vertex for every vertex appearing in the branch. Since each branch has a very small number of vertices compared with the entire network, the vector is typically very sparse. We then construct a covariance matrix from the data of  $B^{max}$  vectors, such that

$$C = V^T V, \text{ where: } \\ V = (\vec{v}_1 - \vec{v}_0) \# (\vec{v}_2 - \vec{v}_0) \# \dots \# (\vec{v}_{B^{max}} - \vec{v}_0) \quad (4) \\ \vec{v}_0 = \frac{1}{B^{max}} \sum_i^{B^{max}} \vec{v}_i$$

Here, we have used  $\#$  to refer to the concatenation of vectors  $\vec{v}_i$  into matrix  $V$ . We can then categorize each vector  $\vec{v}_i$  in  $B^{max}$  according to which eigenvector in  $C$  it is closest to. Most reaction trees obtained from branches up the DFS search tree typically have common 'motifs', associated with one of the principal eigenvalues of  $C$ ; thus, this eigenvalue/eigenvector comparison, or "tree grouping" can be useful to distinguish different 'reaction motifs' from other interesting classes of mechanisms.

## 2.10. Merging More Efficient Branches

Finally, we highlight a further transformation which can be used to simplify CRNs, Travelling up from the root-product, if the sequence of reactions of two sub-branches are identical except for the vertex which gave rise to that sequence (*i.e.* the reactants of the first reaction in that identical sequence are not the same), then we can 'merge' the branches. The assumption underlying this merging is that in some experimental circumstances, one and the same products of some reaction may well encounter themselves reacting again further down the reaction pathway. Figure 6 depicts the merging of two identical branches with vertex  $C$ . We perform this merging for every possible set of branches that can be merged. This means that the total number of leaf-reactants decreases for that reaction tree.



**Figure 6.** The left-hand Figure shows two branches in a reaction-tree which are identical, arising from the same reaction  $A + B \rightarrow C$ , but differing in their parent vertex (in this case,  $A$  and  $B$ , respectively). These branches can be merged together, as shown in the right-hand Figure.

## 2.11. Pseudo-Code for the MechGen Algorithm

We now tie the discussions in Sec. 2.8-2.10 together into a pseudo-code as follows, noting that the leftmost number determines the indentation and the dependency block.

**recursive routine** Get\_Branches(in = vertex molecule  $V^i$ , out = Branches  $B^j$ ):

- 1 -mark  $V^i$  as visited
- 1 if not reached  $G^{max}$ .
- 2 **loop** every reaction edge-tuple  $R_{ij}$
- 3 if not visited edge-tuple before and reaction not exceed  $\tau^{max}$  and all vertices  $V^j$  are unvisited and not reaching  $I^{max}$  when doing so:
- 4 **loop** each of the  $n_e^{R_{ij}}$  vertex molecules  $V^j$
- 5 if  $V^j$  is leaf
- 6 start  $B^j$  by making leaf  $V^j$
- 5 **else**
- 6 -mark  $R_{ij}$  visited
- 6 -Get\_Branches( $V^j, B^j$ )
- 6 -unmark  $R_{ij}$  visited
- 2 -Make all possible  $N_R^{R_{ij}}$  combination of branches  $B^{R_{ij}} = \prod_i^{N_R^{R_{ij}}} B_i$  with  $V^j$  as root
- 2 -Collect all possible branches  $B_i = \sum_j^{N_R^{R_{ij}}} B^{R_{ij}}$
- 2 -Merge together any sub-branches for every branch in  $B_i$
- 2 -Associate every branch with a "tree group" eigenvector
- 2 -Rank every branch
- 1 -unmark  $V^i$  as visited
- 1 -if any, **return**  $B^{max}(1 - \alpha^c)$  of the best and a collection of  $B^{max}\alpha^c$  of the best from each grouping

## 3. Application, Results and Discussion

### 3.1. Calculation Details

The algorithm discussed above allows us to take a large complex CRN, generated by a methodology such as GDS,<sup>[6]</sup> and



then to search this CRN to give a collection of reaction mechanisms (as well as a rough measure of their efficiency) with specific user-defined target products. For each different target product species, our CRN analysis algorithm gives a unique collection of reaction pathways which, if desired, can be further scrutinized using higher-level analysis tools, such as TST rate calculations for the individual reaction steps and more accurate levels of electronic structure.

To test this algorithm, we consider the catalytic oxidation of CO to CO<sub>2</sub> on Pt<sub>*n*</sub> (*n* = 1, 5, 7) clusters. For the two larger size clusters, the cluster structures were taken from the Cambridge Cluster Database.<sup>[52]</sup> For each different cluster size, we performed GDS to generate initial CRNs; these GDS simulations typically comprised the Pt<sub>*n*</sub> cluster, in addition to two CO molecules and a single O<sub>2</sub> molecule. The oxidation of carbon monoxide in the presence of noble metal groups such as platinum have a long history of computational and experimental study, thus serving as useful benchmarks.<sup>[53–57]</sup> We constrained the CRN search by only allowing intermediate species that contain the Pt cluster; in other words, we focus on the surface-catalysed reactivity, rather than gas-phase collisions.

GDS runs, as described in Section 2.2 were used to generate large dense CRNs which were expected to contain all reactions and molecular species relevant to the emergent chemistry of these Pt-catalysed reactions. Here, we note that multiple GDS runs, iteratively starting from different sets of molecular species generated in previous runs, were performed in seeking to exhaustively sample chemical space. To calculate the relative energies of different molecular species, we used the ReaxFF force-field.<sup>[45]</sup> Although not expected to be high-accuracy for general molecular species, ReaxFF has the advantage of being extremely efficient, an important point when dealing with large reaction networks. We note that the force-field parameters employed here are fit to qualitatively reproduce the reactivity of small organic adsorbates (containing C, H and O) undergoing heterogeneous catalysis at Pt and Ni surfaces, and was shown to give qualitatively correct energies of formation of Pt<sub>*x*</sub>O<sub>*x*</sub> (*x* = 1–7) clusters.<sup>[43]</sup> For the purposes of our GDS calculations, the Pt atoms remain fixed at their high-symmetry input geometries, although all Pt are allowed to participate in the reactions. A single CPU was used for a GDS calculation of each Pt cluster, taking about a week of real-time to generate the results.

Because the energies are only expected to be qualitative, as well as wishing to explore as many pathways as possible, we chose a high temperature of 3000 K to estimate the lifetimes. Consequently, the algorithm will have explored far more reaction pathways that realistically necessary at lower temperatures. For the accurate and automated evaluation of the reaction barriers, we use the AutoNEB procedure<sup>[49,50]</sup> in combination with initially-interpolated paths generated using the IDPP potential model.<sup>[58]</sup>

Table 1 gives an indication of the size of the CRNs initially generated by GDS for each Pt cluster, and also summarizes the raw network properties for each cluster. Even for simple reaction systems, it is clear that a large number of molecular species, isomers and associated reactions can be readily generated, motivating the simplifying CRN analyses suggested

**Table 1.** Selected statistics for the Pt<sub>*x*</sub> reaction networks studied here.

Statistic	Pt <sub>1</sub>	Pt <sub>5</sub>	Pt <sub>7</sub>
GDS elementary reactions generated	14541	11989	9669
Molecules in raw network (vertices)	1553	9173	3677
Vertices left after pruning network	845	1405	1161
Stoichiometric reactions (edge-tuples)	6853	8839	7692
Edges left after pruning network	5738	6274	4515
Geometric isomers groups	630	2555	3105
Average number of molecules per isomer group (standard deviation in brackets).	2.5 (3.5)	1.3 (0.9)	1.2 (0.8)

here. Because of the D<sub>h3</sub> and D<sub>h5</sub> symmetries of the larger Pt<sub>5</sub> and Pt<sub>7</sub> clusters, we restricted the GDS exploration by only allowing chemical reactions to occur within a subset of atoms on each cluster, covering adjacent faces of the cluster. This, as well as the fact that these larger clusters also sterically constrain the reaction coordinates available for adsorption, resulted in more reactions found for Pt<sub>1</sub> than for the other two. It also explains why there are substantially more molecules per isomer groups.

We used  $V_{\text{vis}}^{\text{max}} = 10^9$  as the maximum size of exploration; as a result, the graph-search calculations took approximately two days of on a single CPU for all three clusters. The Pt<sub>5</sub> calculations took slightly longer the other two, a result of the large number of sampled edges radiating out from CO<sub>2</sub>. We chose  $G^{\text{max}} = 10$  for this problems, which is sufficiently large to sample a diverse range of reaction mechanisms based on our previous GDS studies.

We calculated the average number of generations searched in the DFS tree before being halted by the parameters  $G^{\text{max}}$ ,  $I^{\text{max}}$  and  $\tau^{\text{max}}$  and we obtained 9.97 (standard deviation 1.12), 9.38 (1.28) and 9.77 (1.19) for the three different cluster sizes. This indicates that the search algorithm was most strongly capped by  $G^{\text{max}}$ , with  $I^{\text{max}}$  and  $\tau^{\text{max}}$  having only limited constraint. The CRN, while large, was therefore almost exhaustively explored up to 10 generations long. This finding also suggests that most of the traversals do not lead to a leaf-reactant below 10 generations; this translates into a large amount of computational time wasted, and its potential consequences/risks discussed in Section 3.4. In our final analysis, we set the maximum number of branches per child vertex  $B^{\text{max}} = 10^4$ , and the proportion of trees from different groups that are passed up the search tree at every generation is  $\alpha^c = 0.05$ .

### 3.2. Results

Here, we discuss the features of the networks and trees generated by our search algorithm, and also highlight some of the low-energy paths for CO oxidation which were generated.

First, Table 2 shows the number of reactions that the molecules CO, O<sub>2</sub> and CO<sub>2</sub> are involved in; these species also happen to be amongst the most connected vertices in the CRNs. For CO<sub>2</sub>, this implies the reaction landscape forms a large “basin of attraction” connected to many less stable molecules,

**Table 2.** The number of elementary reactions (edge-tuples) in which the leaf-reactants (CO and O<sub>2</sub>) and root-product (CO<sub>2</sub>) molecules participate.

Cluster	CO	O <sub>2</sub>	CO <sub>2</sub>
Pt <sub>1</sub>	885	379	645
Pt <sub>5</sub>	2303	221	1831
Pt <sub>7</sub>	2457	242	1571

in agreement with our expectations based on its energetic stability.

From the  $B^{max} = 10^4$  paths generated at the end of our MechGen procedure for each cluster, we are then able to automatically construct the geometries and associated reaction energy profiles for any given reaction-tree extracted from the network; this conversion from graphs to atomic coordinates is achieved using the idea of the GRP, as described in our previous work.<sup>[6,9,14]</sup> After coordinate generation for each reaction intermediate in a given mechanism, we then use the IDPP method to generate initial interpolated reaction-paths, and perform AutoNEB optimization to generate the minimum-energy profile for each elementary reaction; these profiles can be stitched together to give an overview of any of the mechanisms extracted from the initial CRN. Here, we generated such data for the top 500 'fittest' reaction-trees proposed for each cluster.

Under low-pressure experimental conditions, the two main competing mechanisms for CO oxidation are the Langmuir-Hinshelwood (LH) mechanism, where O<sub>2</sub> first dissociates at the Pt surface before reacting with the adsorbed CO to form CO<sub>2</sub>, and the Eley-Rideal (ER) mechanism,<sup>[59]</sup> where CO remains in the gas phase and strikes a chemisorbed O<sub>2</sub>, allowing the product to escape directly into the gas phase. For all three clusters, we found the latter to be prevalent, with most reaction-trees comprising the following three steps (as illustrated in Figure 7):

- Adsorption of O<sub>2</sub> onto the platinum cluster forming Pt<sub>x</sub>O<sub>2</sub>;
- Reaction of the incoming CO with the activated O<sub>2</sub> molecule to form a O<sub>2</sub>-CO intermediate;
- Cleavage of the O-O bond and detachment of CO<sub>2</sub>, leaving behind Pt<sub>x</sub>O.

The ReaxFF forcefield indicates that the O<sub>2</sub> bond-strength is substantially weakened by the unperturbed platinum cluster in all three cases; this weakening of the O<sub>2</sub> bond is in agreement with work by Lu and co-workers.<sup>[60]</sup> In all clusters, we find sites with barrierless adsorption of the O<sub>2</sub> molecule, with a small barrier (<30 kJ mol<sup>-1</sup>) for the formation of the O<sub>2</sub>-CO intermediate and typically a near-barrierless O-O bond breaking, detaching the CO<sub>2</sub> product (Figure 7). The stabilization caused by the O<sub>2</sub> adsorbate onto the Pt<sub>1</sub> atom is four times as great as the two larger clusters, which exhibit very similar profiles. The precise site of absorption, the possible isomerizations of the O<sub>2</sub>-CO intermediates on the surface with low barriers, as well as other minor variations on the ER steps just mentioned, gave rise to the thousands of distinct, albeit similar reaction-trees. These were grouped as described in Sec. 2.9, and we shall describe the results in broad terms for these groups. It is worth noting that our MechGen algorithm also found a number of termolecular barrierless reactions, with the CO, O<sub>2</sub> and Pt<sub>x</sub> reacting

together to concertedly form the O<sub>2</sub>CO intermediate. Though interesting and in some cases could potentially be a mechanism for reaction,<sup>[61]</sup> we exclude these reactions from further discussion.

### 3.2.1. Pt<sub>1</sub>

The prevailing mechanism, representing a tree-group of reactions (as defined in Sec. 2.9) with approximately 59% of all reaction trees found, is in broad agreement with the ER scheme. The O<sub>2</sub> - CO intermediate favours the CO "pointing away" from the Pt atom. This avoids a minima where the CO component of the adsorbed O<sub>2</sub>-CO molecule "wraps around" the Pt atom, subsequently requiring a second barrier to form CO<sub>2</sub>. When the system finds itself in this "wrap around" minima, another mechanism is preferred whereby a second CO attacks the adsorbed O<sub>2</sub>-CO, inducing the formation of CO<sub>2</sub> in a concerted reaction. Variations of this second mechanism, where the CO attacks at different sites of the adsorbed O<sub>2</sub>-CO, are represented by a tree-group comprising ~29% of all trees found.

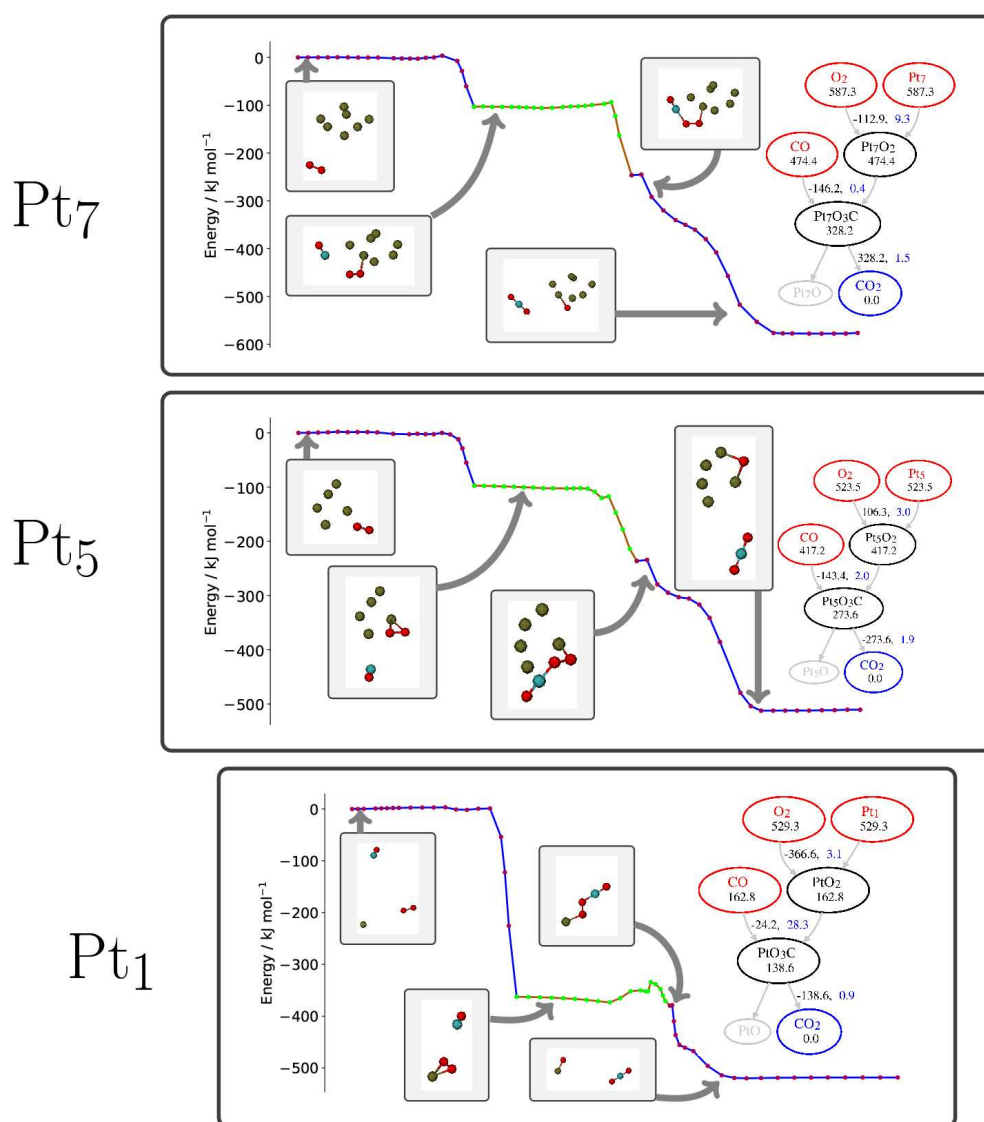
### 3.2.2. Pt<sub>5</sub>

Unlike the single Pt atom, in the case of Pt<sub>5</sub> the CO<sub>2</sub> formation from the adsorbed O<sub>2</sub>-CO is less efficient when CO is pointing outward than when "wrapped around" the cluster. Around 51% of the reaction trees were classified as a ER type, the overwhelming majority of which occur by having the O<sub>2</sub> binding to one or two of the three equatorial Pt atoms. The residual O atom after the CO<sub>2</sub> detaches was not found to preferentially lie on any particular facet, with examples of top, edge and hollow sites all found numerous times within the best 100 reaction paths. Less than 1% of reactions found activated the O<sub>2</sub> at the two Pt atoms above the equatorial plane. A similar set of reactions involved a second CO molecule reacting with the adsorbed O<sub>2</sub>-CO prior to CO<sub>2</sub> formation.

### 3.2.3. Pt<sub>7</sub>

Similar to Pt<sub>5</sub>, the ER mechanism was overwhelmingly favoured in our CRN, with most O<sub>2</sub> preferentially binding to one of the five planar symmetric Pt atoms, or on the edge between these and the two symmetric Pt atoms. Only a small number of reaction pathways involved binding solely to the two symmetric Pt atoms. The adsorbed O<sub>2</sub>-CO molecule is found preferentially pointing away from the cluster (as shown in Figure 7), but sometimes "wrapping onto" the top two symmetric Pt atoms. The residual oxygen atom is typically found at either hollow or edge sites.

For both the Pt<sub>5</sub> and Pt<sub>7</sub> clusters, there are many reaction mechanisms with low barriers which also include a second CO attached to the surface at different sites. The barrier for these reactions appear most of the time to follow a similar mechanism as without co-adsorption of the second CO,



**Figure 7.** Energy profiles for the 'best' (*i.e.* lowest energy barrier) reaction-trees for clusters Pt<sub>1</sub> (bottom), Pt<sub>5</sub> (middle) and Pt<sub>7</sub> (top). All energies are in kJ mol<sup>-1</sup>. The bubble diagrams depict the reaction-tree producing CO<sub>2</sub>; for all three the mechanism this broadly corresponds to the Eley-Rideal (ER) mechanism. Within each vertex, the number displays the system energy before the elementary reaction takes place. The two numbers associated with each edge-tuple are the change in internal energy for that reaction (in black) and the activation energy barrier relative to the initial state of that elementary reaction (in blue).

suggesting that the ReaxFF force field may not be accurate to describe the perturbation to the PES surface caused by these secondary CO molecules. Overall, the similarity of all three energy profiles for the best reaction mechanism for each cluster is encouraging in suggesting that the MechGen algorithm working, but might also suggest that ReaxFF is not sensitive enough to capture the differences in the size effects of these.

### 3.3. Using BEP Barriers

The calculation of activation energies for tens of thousands of reactions may well require unreasonable computational resources, even when using reactive force-fields such as ReaxFF. Here, we briefly explore the possibility of using barriers estimated by

a single-parameter BEP approach ( $\Delta E_i = \alpha \Delta U_i$ ) as a means of pre-screening reaction mechanisms before calculating their activation energies. To this end, we ran the same MechGen calculations for the three input CRNs for different Pt clusters, as presented above, but used BEP-estimated barriers, instead of the more accurate, ReaxFF barriers, in estimating efficiencies. We emphasize here that our application of the BEP principle is very approximate, in the sense that we have not optimized the  $\alpha$  parameter independently for each reactant species or reaction class.

In all cases, the resulting mechanisms suggested when using the BEP approach were different in appearance to those extracted using ReaxFF, often suggesting mechanisms with multiple isomerization steps before CO<sub>2</sub> formation. Nevertheless, the broad sequence of steps also generally followed the ER

mechanism, albeit in a more circuitous manner than the mechanisms obtained with ReaxFF barriers. We thus conclude that using the BEP barriers in the present for is insufficient on its own, within this algorithm, to locate the most sensible mechanisms for a reaction.

However, it is also instructive to analyse these results in a more rigorous manner, by comparing the statistical grouping using the covariance matrix eigen-decomposition described in section 2.9. Here, we calculated the overlap of the eigenvectors for the different mechanisms groupings generated with BEP-estimated barrier to those estimated with ReaxFF. The top ten eigenvectors of each calculation span >90% of the vertex space in the reaction tree datasets. The length of the first five normalised eigenvectors of the ReaxFF calculation, orthogonally projected onto the space of the top ten eigenvectors of the BEP calculation, are shown in Table 3; the rightmost column shows the fraction of overlap between the top ten eigenvectors. These results, particularly the final column in Table 3, show that about half of the vertices visited by both methods overlap. Notably, the first eigenvector for all three clusters in the ReaxFF calculation very strongly overlaps with the first of the BEP calculation (as shown by the first column of Table 3), highlighting the similarity in the most common mechanisms found by both ReaxFF and the BEP approach.

### 3.4. Discussion

Despite the successful application of this algorithm to uncover physically-sensible mechanisms in these test systems, there are a number of limitations and improvements that we shall now discuss.

**Missing reactions:** The first “red flag” worth discussing is lack of the LH mechanism appearing in proposed mechanisms for any of the three clusters. The absence of the LH mechanism, thought to be an important mechanism for the formation of CO<sub>2</sub> on surfaces and large nanoclusters,<sup>[62,63]</sup> may well simply be explained by the very small size of the Pt clusters, which might not permit O<sub>2</sub> to dissociate on the cluster ‘surface’ while simultaneously permitting adjacent adsorption of CO. The near-barrierless ER mechanism found also raises questions about the qualitative accuracy of the ReaxFF potential. Here, we double-checked our results to ensure that the climbing-image routine in the NEB algorithm did not ‘skip’ the TS due to limited number of images. The near-barrierless mechanism seems plausible in this case, since the atoms forming CO<sub>2</sub> were not themselves strongly bound to the cluster, and the mechanism avoids the binding of CO or atomic O on the Pt surface,

adsorbates which typically exhibit strong binding to Pt clusters (binding strength of the adsorbates O > CO > O<sub>2</sub> > CO<sub>2</sub><sup>[66]</sup>).

It may also be the case that the LH mechanism barrier is overestimated by the ReaxFF forcefield and exceeds the  $\tau^{max}$  threshold; it has been argued that for Pt(111), the reaction barriers are higher for the LH mechanism compared to that of ER.<sup>[62]</sup> If the ReaxFF forcefield deems the barrier of such a reaction too high, the MechGen algorithm will consequently discriminate against reaction trees containing elementary steps associated with the LH mechanism. However, despite the higher barrier, the LH mechanism might still conceivably be the prevalent route for formation of CO<sub>2</sub>,<sup>[62]</sup> and so it would obviously be desirable for the MechGen algorithm to uncover such reaction pathways.

**Too many branches exceeding  $G^{max}$  in the DFS tree waste CPU-resources:** As briefly touched upon in Section 3, the average number of generations (*i.e.* the length of the branches) of the DFS tree searched was, on average, close to the maximum allowed number of generations in the search  $G^{max}$ . This suggests that most of the searched trees will be longer than  $G^{max}$  before they find a reaction-leaf; evaluating a large number of such branches in the DFS tree is clearly a waste of CPU-resources. As discussed above,  $\tau^{max}$  is chosen to produce a target value of  $V_{vis}^{max}$ ; if a large proportion of the visited vertices are low barrier edges that do not lead to a reaction-leaf in fewer than  $G^{max}$  generations, the chosen  $\tau^{max}$  may filter-out complete reaction trees which might have led to given leaf-reactants for slightly large  $G^{max}$ . It is therefore conceivable that the LH mechanism, with fewer than  $G^{max}$  steps, may have been screened out, owing to the ReaxFF estimated barrier height being higher than many small barrier traversals that do not reach a leaf-reactant in less than  $G^{max}$  generations. Approaches to potentially cutting such dead-end branches of the reaction-trees will be sought in future developments.

**Improvements on the classification and diversification of groups of reaction-trees:** Even for the simple systems considered here, the extracted mechanisms already show a large number of variations on a single ‘mechanistic theme’. Despite taking point group and permutational symmetry into consideration, the precise angle the adsorbate makes with the cluster, the minor variations of the O<sub>2</sub>–CO intermediate, the precise position with respect to the Pt atom site, and many other factors, all serve to multiply the number of distinct reaction trees, albeit broadly with the same mechanisms. As a result, there is still a demand to find a way of clearly classifying and identifying as many diverse mechanistic themes as possible. To a reasonable extent, the current implementation captures these variations and groups them into the principal components of the covariance matrix of the reaction tree population, as

**Table 3.** The norm of the first five eigenvectors of the ReaxFF calculation in the space of the first ten eigenvectors of the BEP calculation. The last column shows the space of overlap between the top ten eigenvectors for both calculations.

Clu	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$\frac{1}{10} \sum_{i=1}^{10} v_i$
Pt <sub>1</sub>	0.98	0.30	0.78	0.85	0.38	0.45
Pt <sub>5</sub>	0.97	0.52	0.30	0.83	0.43	0.51
Pt <sub>7</sub>	0.98	0.34	0.63	0.23	0.61	0.58



described in Section 2.9. Nevertheless, the classification seems to be too broad, and it does not rely on geometric relationships as part of the classification, only on energy profiles. Strategies for better classification of reaction trees will be an important further development of this algorithm; the aim will be to reduce the number of repetitive variations of the same 'mechanism theme', and to give a more broad and diverse collection of distinct reaction mechanisms. Approaches like these could bias the algorithm towards finding as many plausible 'mechanism themes' as possible, for example by classifying mechanisms based on principal components of the covariance of the population of mechanisms. Such approaches may then help in subsequently using higher levels of theory for the evaluation of barriers to conclusively determine the correct mechanism and remove any bias which may emerge from the potential energy method itself.

**BEP barriers as a first-pass screening:** It is somewhat surprising that the BEP and ReaxFF results have such an overlap for the resulting eigenvectors, especially as the ReaxFF model suggests a low barrier for O–O bond breaking, typically having a large change in internal energy  $\Delta U_i$ . The fact that these distinct sets match could simply be due to the connective nature of the networks generated, restricted by design to particularly explore the formation of CO<sub>2</sub>. It could also merely reflect the large number of similar variations/combinations of different isomerization available for the same 'mechanism theme' found, since the principal components of the covariance matrix in Eq. 4 are provided by the frequency of reaction trees in the dataset. A unique, albeit crucial mechanism with few variations on the theme (few isomerization edges connecting to it) will probably have a low contribution on the construction of the covariance matrix, and would risk being missed by using this procedure. Nevertheless, future work will continue to analyse this potential avenue for drastically pruning the number of vertices in the network, in particular using more accurate estimations of the BEP barriers by using parameters fitted to experimental or computational data.<sup>[65]</sup>

## 4. Conclusions

This article has outlined a novel strategy for generating a large collection of efficient reaction mechanisms, represented by tree-like graphs connecting some pre-selected set of reactants which, via a sequence of elementary reactions, can arrive at a desired preselected product. The scheme is relatively simple and as such could be easily used as a start for even more sophisticated searching strategies along similar lines. Furthermore, we note that this approach is generally applicable to analysis of CRNs generated using any procedure for constructing kinetic networks, and is also compatible with any method for calculating energetics of the constituent reactions. The test systems investigated here show that, from a database of more than ten thousand reactions, the relevant intermediate molecules can be reduced to a smaller CRN with approximately a thousand vertices and five thousand edges; we have also highlighted the extraction of the 'most likely' reaction-mechanisms

from this reduced CRN using DFS-based algorithms coupled to sensible search-restriction criteria.

For the cases considered here, it seems likely that the minimum-energy mechanisms were found; in particular, based on the ReaxFF force field, near-barrierless elementary reactions were found in the key proposed mechanisms. Furthermore, the mechanism broadly agree with the Eley-Rideal mechanism, where the CO reacts with the adsorbed O<sub>2</sub> to subsequently detach the CO<sub>2</sub> without itself adsorbing into the cluster face. The algorithm generated thousands of variation of the ER theme, with a few other mechanisms, notably a second CO inducing the formation of CO<sub>2</sub> in different ways. The Langmuir-Hinshelwood mechanism was notably absent in our searches, but this might simply reflect limitations placed on this mechanism in the case of such small systems. However, a number of improvements are left to make this algorithm more effective at classifying mechanism families and reducing redundancies on the network that threaten to make the algorithm inefficient, and these were also discussed. Nevertheless, this algorithm and overall methodology is already reliable enough to be tested using more realistic cluster sizes and more accurate electronic structure calculations; improved parallelization of the GDS procedure will enable more exhaustive exploration of variable defect sites in larger clusters (e.g. hundreds of Pt atoms) using more accurate semiempirical methods such as DFT Tight-Binding.<sup>[65]</sup> This network representation and searching methodology is leading to new avenues for further algorithmic developments which may illuminate real problems, such as understanding the size-dependency of the catalytic activity of Pt clusters, studies which are already under way.

## Data availability

Data and molecular structure for Figure 7 are available at [wrap.warwick.ac.uk/126171](http://wrap.warwick.ac.uk/126171)

## Acknowledgements

The authors gratefully acknowledge the Engineering and Physical Sciences Research Council (EPSRC) for award EP/R020477/1, and the Scientific Computing Research Technology Platform at the University of Warwick for providing computational resources.

## Conflict of Interest

The authors declare no conflict of interest.

**Keywords:** Catalysis · graph searching algorithms · nudged-elastic-band · reaction discovery · reaction mechanism

[1] T. Turanyi, A. S. Tomlin, *Analysis of kinetic reaction mechanisms*, Springer 2014.



- [2] P. van Leeuwen, N. M. W., *Homogeneous Catalysis: Understanding the Art*, Kluwer Academic Publishers **2004**.
- [3] K. J. Laidler, *Chemical Kinetics* 3rd ed., Harper Collins: New York **1987**.
- [4] E. Martínez-Núñez, *J. Comput. Chem.* **2015**, *36*, 222–234.
- [5] K. Ohno, S. Maeda, *Phys. Scr.* **2008**, *78*, 058122.
- [6] S. Habershon, *J. Chem. Theory Comput.* **2016**, *12*, 1786–1798.
- [7] C. F. Goldsmith, R. H. West, *J. Phys. Chem. C* **2017**, *121*, 9970–9981.
- [8] C. A. Class, M. Liu, A. G. Vandeputte, W. H. Green, *Phys. Chem. Chem. Phys.* **2016**, *18*, 21651–21658.
- [9] I. Ismail, H. Stuttaford-Fowler, V. A. Ochan Ashok, B. C. Robertson, S. Habershon, *J. Phys. Chem. A* **2019**, *123*, 3407–3417.
- [10] L.-P. Wang, A. Titov, R. McGibbon, F. Liu, V. S. Pande, T. J. Martínez, *Nat. Chem.* **2014**, *6*, 1044–8.
- [11] G. N. Simm, A. C. Vaucher, M. Reiher, *J. Phys. Chem. A* **2019**, *123*, 385–399.
- [12] S. Maeda, K. Ohno, *J. Phys. Chem. A* **2005**, *109*, 5742–5753.
- [13] A. L. Dewyer, A. J. Argüelles, P. M. Zimmerman, *WIREs Comput. Mol. Sci.* **2018**, *8*, e1354.
- [14] S. Habershon, *J. Chem. Phys.* **2015**, *143*, 094106.
- [15] S. Maeda, K. Morokuma, *J. Chem. Theory Comput.* **2012**, *8*, 380–385.
- [16] M. Keçeli, S. N. Elliott, Y.-P. Li, M. S. Johnson, C. Cavallotti, Y. Georgievskii, W. H. Green, M. Pelucchi, J. M. Wozniak, A. W. Jasper, S. J. Klippenstein, *Proc. Combust. Inst.* **2019**, *37*, 363–371.
- [17] C. W. Gao, J. W. Allen, W. H. Green, R. H. West, *Comp. Phys. Comm.* **2016**, *203*, 212–225.
- [18] P. M. Zimmerman, *J. Comput. Chem.* **2013**, *34*, 1385–1392.
- [19] A. J. Nett, W. Zhao, P. M. Zimmerman, J. Montgomery, *J. Am. Chem. Soc.* **2015**, *137*, 7636–9.
- [20] P. M. Zimmerman, *J. Chem. Theory Comput.* **2013**, *9*, 3043–3050.
- [21] G. N. Simm, M. Reiher, *J. Comp. Theory Comput.* **2017**, *13*, 6108–6119.
- [22] J. Proppe, M. Reiher, *J. Comp. Theory Comput.* **2019**, *15*, 357–370.
- [23] Y. Kim, S. Choi, W. Y. Kim, *J. Chem. Theory Comput.* **2014**, *10*, 2419–2426.
- [24] Y. Kim, J. W. Kim, Z. Kim, W. Y. Kim, *Chem. Sci.* **2018**, *9*, 825–835.
- [25] C. W. Coley, W. Jin, L. Rogers, T. F. Jamison, T. S. Jaakkola, W. H. Green, R. Barzilay, K. F. Jensen, *Chem. Sci.* **2019**, *10*, 370–377.
- [26] J. S. Schreck, C. W. Coley, K. J. M. Bishop, *ACS Cent. Sci.* **2019**, *5*, 970–981.
- [27] W. Jin, C. Coley, R. Barzilay, T. Jaakkola, *Advances in Neural Information Processing Systems*, **2017**, 2607–2616.
- [28] P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. Bekas, A. A. Lee, *ChemRxiv* **2019**.
- [29] J. A. Varela, A. S. Vázquez, E. Martínez-Núñez, *Chem. Sci.* **2017**, *8*, 3843–3851.
- [30] A. Rodríguez, R. Rodríguez-Fernández, A. S. Vázquez, E. Martínez-Núñez, *J. Comb. Chem.* **2018**, *39*, 1922–1930.
- [31] D. G. Truhlar, B. C. Garrett, S. J. Klippenstein, *J. Phys. Chem.* **1996**, *100*, 12771–12800.
- [32] K. J. Laidler, M. C. King, *J. Phys. Chem.* **1983**, *87*, 2657–2664.
- [33] N. E. Henriksen, F. Y. Hansen, *Theories of Molecular Reaction Dynamics: The Microscopic Foundation of Chemical Kinetics*, Oxford University Press **2011**.
- [34] D. T. Gillespie, *J. Comp. Physiol.* **1976**, *22*, 403–434.
- [35] D. T. Gillespie, *J. Phys. Chem.* **1977**, *81*, 2340–2361.
- [36] D. T. Gillespie, A. Hellander, L. R. Petzold, *J. Chem. Phys.* **2013**, *138*, 170901.
- [37] M. Besora, F. Maseras, *WIREs Comput. Mol. Sci.* **2018**, *8*, e1372.
- [38] X. Chen, C. F. Goldsmith, *J. Phys. Chem. A* **2017**, *121*, 9173–9184.
- [39] C. W. Gao, J. W. Allen, W. H. Green, R. H. West, *Comp. Phys. Comm.* **2016**, *203*, 212–225.
- [40] F. Perini, E. Galligani, R. D. Reitz, *Energy Fuels* **2012**, *26*, 4804–4822.
- [41] C. K. Westbrook, W. J. Pitz, O. Herbinet, H. J. Curran, E. J. Silke, *Combust. Flame* **2009**, *156*, 181–199.
- [42] D. Fantauzzi, J. Bandlow, L. Sabo, J. E. Mueller, A. C. van Duin, T. Jacob, *Phys. Chem. Chem. Phys.* **2014**, *16*, 23118–23133.
- [43] Y. K. Shin, L. Gai, S. Raman, A. C. van Duin, *J. Phys. Chem. A* **2016**, *120*, 8044–8055.
- [44] M. Buehler, A. Duin, W. Goddard, T. Jacob, Y. Jang, B. Merinov, *Mater. Res. Soc. Symp. Proc.* **2005**.
- [45] A. C. Van Duin, S. Dasgupta, F. Lorant, W. A. Goddard, *J. Phys. Chem. A* **2001**, *105*, 9396–9409.
- [46] M. G. Evans, M. Polanyi, *Trans. Faraday Soc.* **1938**, *34*, 11–24.
- [47] R. A. van Santen, M. Neurock, S. G. Shetty, *Chem. Rev.* **2009**, *110*, 2005–2048.
- [48] A. Logadottir, T. H. Rod, J. K. Nørskov, B. Hammer, S. Dahl, C. Jacobsen, *J. Catal.* **2001**, *197*, 229–231.
- [49] E. L. Kolsbjerg, M. N. Groves, B. Hammer, *J. Chem. Phys.* **2016**, *145*, 094107.
- [50] H. Jónsson, G. Mills, K. W. Jacobsen, *Citeseer* **1998**.
- [51] S. Even, *Graph Algorithms*, Cambridge University Press **2011**.
- [52] D. Wales, J. Doye, A. Dullweber, M. Hodges, F. Naumkin, F. Calvo, J. Hernandez-Rojas, T. Middleton, *The Cambridge Cluster Database* <http://www.wales.ch.cam.ac.uk> **2005**.
- [53] A. Smeltz, R. Getman, W. Schneider, F. Ribeiro, *Catal. Today* **2008**, *136*, 84–92.
- [54] J. Bray, W. Schneider, *Langmuir* **2011**, *27*, 8177–8186.
- [55] D. J. Schmidt, W. Chen, C. Wolverton, W. F. Schneider, *J. Chem. Theory Comput.* **2011**, *8*, 264–273.
- [56] M. Stamatakis, D. G. Vlachos, *ACS Catal.* **2012**, *2*, 2648–266.
- [57] Q. Fu, J. Yang, Y. Luo, *J. Phys. Chem. C* **2011**, *115*, 6864–6869.
- [58] S. Smidstrup, A. Pedersen, K. Stokbro, H. Jónsson, *J. Chem. Phys.* **2014**, *140*, 214106.
- [59] D. Eley, E. Rideal, *Nature* **1940**, *146*, 401.
- [60] Y. Lu, J. Wang, L. Yu, L. Kovarik, X. Zhang, A. S. Hoffman, A. Gallo, S. R. Bare, D. Sokaras, T. Kroll, V. Dagle, H. Xin, A. M. Karim, *Nature Catal.* **2019**, *2*, 149–156.
- [61] M. P. Burke, S. J. Klippenstein, *Nat. Chem.* **2017**, *9*, 1078.
- [62] R. Baxter, P. Hu, *J. Chem. Phys.* **2002**, *116*, 4379–4381.
- [63] I. Langmuir, *Trans. Faraday Soc.* **1922**, *17*, 621–654.
- [64] S. Wang, V. Petzold, V. Tripkovic, J. Kleis, J. G. Howalt, E. Skulason, E. M. Fernandez, B. Hvolbaek, G. Jones, A. Toftlund, H. Falsig, M. Bjorketun, F. Studt, F. Abild-Pedersen, J. Rossmeisl, J. K. Nørskov, T. Bligaard, *Phys. Chem. Chem. Phys.* **2011**, *13*, 20760–20765.
- [65] S. Grimme, C. Bannwarth, P. Shushkov, *J. Chem. Theory Comput.* **2017**, *13*, 1989–2009.
- [66] Y. Xu, R. B. Getman, W. A. Shelton, W. F. Schneider, *Phys. Chem. Chem. Phys.* **2008**, *10*, 6009–6018.

Manuscript received: September 27, 2019  
Version of record online: December 17, 2019