
Article:
Flamm, C, Hellmuth, M, Merkle, D et al. (2020) Generic Context-Aware Group Contributions. IEEE/ACM Transactions on Computational Biology and Bioinformatics. ISSN: 1545-5963

https://doi.org/10.1109/tcbb.2020.2998948

Generic Context-Aware Group Contributions

Christoph Flamm, Marc Hellmuth, Daniel Merkle, Nikolai Nøjgaard, Peter F. Stadler

Abstract—Many properties of molecules vary systematically with changes in the structural formula and can thus be estimated from regression models defined on small structural building blocks, usually functional groups. Typically, such approaches are limited to a particular class of compounds and requires hand-curated lists of chemically plausible groups. This limits their use in particular in the context of generative approaches to explore large chemical spaces. Here we overcome this limitation by proposing a generic group contribution method that iteratively identifies significant regressors of increasing size. To this end, LASSO regression is used and the context-dependent contributions are "anchored" around a reference edge to reduce ambiguities and prevent overcounting due to multiple embeddings. We benchmark our approach, which is available as "Context AwaRe Group cOntribution" (CARGO), on artificial data, typical applications from chemical thermodynamics. As we shall see, this method yields stable results with accuracies comparable to other regression techniques. As a by-product, we obtain interpretable additive contributions for individual chemical bonds and correction terms depending on local contexts.

Index Terms—Group Contributions, Thermodynamics, Lasso regression, Frequent Subgraph Mining, Cheminformatics

1 Introduction

Group contribution methods have a long history in chemistry. They are based on the observation that simple constituents of chemical structures, often identified as functional groups, contribute in a predictable and largely context-independent way to a molecules physico-chemical properties. In the simplest case, group contributions are additive and we will only be interested with additive models in this contribution.

Many thermodynamic properties can be estimated in this manner from the molecular structure alone. In classical approaches such as the Joback/Reid method, molecules are partitioned into (functional) groups each of which contributes a single term [1]. This simple ansatz yields useful estimates e.g. for boiling and melting points, critical temperature and pressure, heat of formation, Gibbs energy of formation. More elaborate schemes, such as UNIFAC [2] in addition model interactions between groups at the expense of a much larger set of parameters.

From a mathematical point of view, group contribution methods are regression models. Despite their differences in the details, they share a common strategy comprising three distinct steps [3]: (i) collection of a database of accurate experimental data, from which the parameters of the group contribution are learned, (ii) identification of the (functional) groups, and (iii) a decomposition of molecules into their constituents.

Step (ii) usually involves manually determined lists of groups that are tailored to the specific chemistry being modelled. The need for pre-defined groups, however, becomes a problem when very large and diverse sets of compounds are of interest. In the context of generative models of chemistry such as MØD [4], [5], this limitation becomes crippling and limits the exploration of chemical space to compounds that are entirely composed of the predetermined groups. In order to overcome this limitation, we propose here a graph-theoretical approach to defining groups that is agnostic of chemical knowledge but nevertheless encodes the salient features implicitly in a manner that still allows a direct interpretation of the regression model.

Traditionally, group contribution methods often require in step (iii) that the functional groups form a vertex partition. This kind of tiling problem is known to be NP-complete [6]. More problematically, however, there is no guarantee that the partition is unique. As a consequence, both parameter estimation in the training step and the estimation of properties become dependent on the ambiguous partition, adding a layer of avoidable inaccuracy. We therefore abandon the requirement of vertex partitoning here and instead opt for a graph covering scheme that is guaranteed to be unambiguous. In some application scenarios, vertex partitions even are not desirable from a chemist's point of view: Molecular energies, for instance, are usually very well approximated as the sum of the average bond energy for each chemical bond in the molecule. The energy contributions are parametrized as a function of the two incident atoms and the type of the chemical bond [7], [8]. Modelling energies more accurately at the level of quantum mechanics we see, however, that the total energy appears as the solution of a large eigenvalue problem that does not factorize exactly. Hence there is also no partition of the molecule into parts that contribute strictly additively. The localization of electrons in discernible bonds, on the other hand, ensures that contribution of individual bonds are almost always an acceptable approximation.

Nevertheless, the chemical environment of a bond has a non-negligible influence. For instance C—C bonds have somewhat different average energies depending on whether one or both carbons is incident to a single or double bond, or

- *CF, PFS: Institute for Theoretical Chemistry, University of Vienna, Wien A-1090, Austria*
- *MH: School of Computing, University of Leeds, LS2 9JT Leeds, UK*
- *DM: Department of Systems Biology, Harvard Medical School, 200 Longwood Avenue, Boston, MA 02115 USA*
- *NN, DM: Department of Mathematics and Computer Science, University of Southern Denmark, Odense M DK-5230, Denmark*
- *PFS: Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics, University of Leipzig, Leipzig D-04107, Germany*

part of a carbonyl group. Clearly, if we allow the contribution of a bond to depend on the entire rest of the molecule, this additivity of the contributions is a mathematical triviality (and in fact the decomposition into bond contributions is arbitrary). The approach becomes non-trivial (and practically useful) only if it is possible to limit the size of a bond's context sufficiently to enumerate all relevant contexts and to assign energies to them. Details of the implementation vary depending on the scope of chemistry and the properties under consideration. In [9], [10] for instance, contributions are divided into first, second, and third order groups. An interesting extension is the estimation of standard Gibbs energy of reactions combining reactant contributions and group contributions [11].

Group contribution models are by no mean specific to chemistry or molecular properties. The prediction of RNA secondary structure, for example, relies on the fact that additive contributions of base-pair stacking and loop strain energies explain measured free energies of folding with very high accuracy [12]. This example provides further motivation for the construction of a group contribution method that is applicable generically to labeled graphs (c.f. Supplemental Information).

2 Notation

In this contribution we consider undirected simple graphs that are equipped with an edge- and vertex-labeling, henceforth called *graphs* for short. The vertex set and edge set of a graph G is denoted by $V(G)$ and $E(G)$, respectively. Moreover, we write $l(x)$ for the vertex-labels ($x \in V(G)$) and edge-labels ($x \in E(G)$). Since our approach is strongly motivated by chemistry, sometimes it is more natural to talk about the graphs as molecules, their vertices as atoms (with labels defining the atom type), and their edges as bonds (whose labels distinguish single, double, triple, and aromatic bonds, for instance), while still using common graph terminology for mathematical precision.

Given two graphs G and G' and a bijection $\varphi : V(G) \to V(G')$, we say φ is *edge preserving* if $(v,u) \in E(G)$ if and only if $(\varphi(v),\varphi(u)) \in E(G')$. Moreover, an edge preserving map φ is *label preserving* if for any $v \in V(G)$ it holds that $l(v) = l(\varphi(v))$ and for any edge $(v,u) \in E(G)$ we have $l((v,u)) = l((\varphi(v),\varphi(u)))$. The bijection φ is called an *isomorphism* if it is both edge and label preserving, and we say that G and G' are isomorphic, denoted by $G \simeq_\varphi G'$, if there exists an isomorphism φ between them. If there is no risk of ambiguity or the map φ is not important, we simply write $G \simeq G'$. If $G = G'$, then any isomorphism between G and G' is also called an *automorphism*.

A graph H is a *subgraph* of G, in symbols $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. Given two graphs H' and G we say that H' is *subgraph isomorphic* to G, if there exists a subgraph H of G with $H' \simeq H$. The isomorphism φ between H' and a subgraph $H \subseteq G$ is called a *subgraph isomorphism between H' and G* and, by slight abuse of notation, we also write $H' \subseteq G$.

3 Problem Definition

Given a sample $\mathcal{S} \subseteq \mathcal{G}$ of pairwisely distinct graphs drawn from a target set $\mathcal{G}$, we assume that the value $t_{\text{obs}}(S)$ of a quantity of interest has been measured for all $S \in \mathcal{S}$. Our goal is to train a regression model for t_{obs} that extends to all elements of $\mathcal{G}$. The model t belongs to a restricted class of functions $t : \mathcal{G} \to \mathbb{R}$, in our case functions that are linear combinations of contributions of certain subgraphs. The regressions model t of course has to be an approximation of t_{obs} on $\mathcal{S}$. This setup can formalized as follows:

Problem 1. `GraphRegression`
Instance: *A set of graphs $\mathcal{G}$, a subset of pairwisely non-isomorphic graphs $\mathcal{S} \subseteq \mathcal{G}$, a class of functions $\mathbb{F} \subseteq \mathbb{F}_0 := \{t \mid t : \mathcal{G} \to \mathbb{R}\}$, and a performance measure $\Delta : \mathbb{F}_0(\mathcal{S}) \times \mathbb{F}_0(\mathcal{S}) \to \mathbb{R}_0^+$ where $\mathbb{F}_0(\mathcal{S})$ is the restriction of $\mathbb{F}_0$ to $\mathcal{S}$.*
Task: *Find $t \in \mathbb{F}$ such that $\Delta(t, t_{obs})$ is minimal.*

In the following section we will specify in detail the class of subgraph-additive functions on graphs and then argue for a further restriction of these functions depending on the subgraphs actually observed in the training set $\mathcal{S}$. The class of regression functions $\mathcal{F}$ advocated here will comprise linear functions of subgraphs that are observed "significantly" in $\mathcal{S}$. As performance measures we will use the well-established LASSO operator [13], see section 4.1.4.

4 Context-Aware Group Contributions

4.1 Context-dependent Edge Contributions

4.1.1 Definitions

We start from the set $\mathbb{E} = \bigcup_{S \in \mathcal{S}} E(S)$ of all edges in the training set $\mathcal{S}$ and introduce a map $t_{\text{edge}} : \mathbb{E} \to \mathbb{R}$ such that

$$t_{\text{obs}}(S) = \sum_{e \in E(S)} t_{\text{edge}}(e). \tag{1}$$

At the outset, of course we do not know the values of t_{edge}. Eq. (1) merely makes our notion of additive contributions precise. The key assumption is that $t_{\text{edge}}(e)$ is determined by the surrounding structural context of the graph in which e resides. Although the ansatz of Eq. (1) is motivated by bond energy contributions, it is much more general and pertains to many other properties including those covered by the Joback/Reid method, see Sec. 6 below.

Next, we formalize the notion of *context around an edge*.

Definition 1. A *context* is a pair $C = (G, e)$, where G is a graph and e is an edge in G. The *size of C* is defined as the number of edges in G. We call e the *origin* or *reference edge* of $C = (G, e)$. The context C of size one, consisting of a single edge, is called *trivial*.

Two contexts $C_1 = (G_1, e_1)$ and $C_2 = (G_2, e_2)$ are *isomorphic*, in symbols $C_1 \simeq {}_\varphi C_2$, if there exists an isomorphism φ from G_1 to G_2 that maps e_1 to e_2. If $C_1 \simeq_\varphi C_2$, $G_1 = G_2$ and $e_1 = e_2$, we write $C_1 = C_2$ and say that φ is an *automorphism*.

Two contexts $C_1 = (G_1, e_1)$ and $C_2 = (G_2, e_2)$ are *subgraph isomorphic*, in symbols $C_1 \sqsubseteq C_2$ if there exists a subgraph H of G_2 such that there is an isomorphism from G_1 to H that maps e_1 to e_2. If $C_1 \sqsubseteq C_2$ we also say that C_1 can be *embedded into* C_2.

In what follows we adopt the convention of coloring the reference edge red for any illustrated context. Intuitively, the context is a local substructure around a certain edge

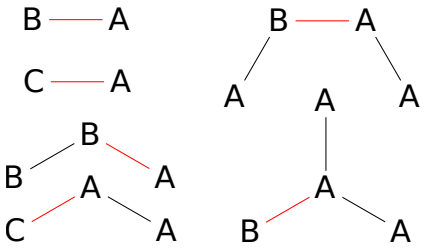


Fig. 1. Depicted are different contexts of the sizes 1 (trivial), 2, and 3. The origin edge of each context is colored red.

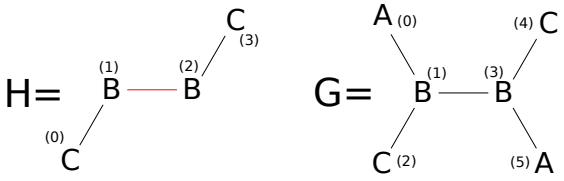


Fig. 2. Shown are two graphs H (left) and G (right) together with a vertex-label and an index of each vertex (the number in brackets). For simplicity all edge labels are the same and are omitted in the drawing. The graph H depicts a context $C = (H, e)$ with the origin edge $e = (1, 2)$ which is highlighted by a red edge. The graph G depicts a sample graph in which we want to determine the frequency of C. For any edge $e' \in E(H)$ with $e' \neq (1, 3)$ we have $f(C, G, e') = 0$. Furthermore, $f(H, G, e') = 2$, since there exists two subgraph isomorphisms from H to G: one that maps the vertex 0 of H to the vertex 2 in G and one that maps the vertex 0 of H to the vertex 4 in G.

in a graph. More precisely, given a graph G and a context $C = (H, e)$ such that there is a subgraph isomorphism φ between H and G, then H can be seen as a context around $\varphi(e)$. We define the number of occurrences of C around an edge in G as follows:

Definition 2. Given a graph G and a context $C = (H, e')$ we say that C is *a context around* $e \in E(G)$, if there is a subgraph isomorphism φ from H to G that maps the origin edge to e, i.e., it satisfies $\varphi(e') = e$. The *frequency* $f(C, G, e)$ of C around some edge $e \in E(G)$ is the number of subgraph isomorphisms $\varphi_1, \varphi_2, \ldots$ from C to G that satisfy $\varphi_i(e') = e$.

An example of counting the frequency of a context is shown in Fig. 2.

We can similarly define the total number of occurrences of C in G as follows:

Definition 3. Given a graph G and a context C, the *frequency* of C in G is defined as the sum of the frequencies of C around all edges in G:

$$f(C, G) = \sum_{e \in E(G)} f(C, G, e).$$

We emphasize that C can be embedded into G if and only if $f(C, G) > 0$.

The definition of the frequency of a context in a graph does not account for the presence of symmetries. That is, the context in Fig. 2 is counted twice in the given example, while it might be more intuitive to count such occurrences only once. In general, if $a(C)$ is the number of automorphisms of a given context C, then we will "over-count" the context around an edge $a(C)$ times, as for the edge $e = (1, 3)$ in the example of Fig. 2. It would easily be possible to correct this "over-counting" by dividing the frequency $f(C, G)$ by $a(C)$. It is not necessary, however, to account for automorphisms of C since they are independent of the embedding of C into G. As we shall see below, the frequencies $f(C, G)$ will appear as coefficients of regressors in a linear regression model, hence a factor that is independent from G will be canceled by corresponding scaling of the regressor.

The set of non-isomorphic contexts that are embeddable into the graphs $S \in \mathcal{S}$ is finite. We define this set as follows:

Definition 4. The context set $\mathfrak{C}_k^{\mathcal{S}}$ is the set of all contexts of size k, such that for each context $C \in \mathfrak{C}_k^{\mathcal{S}}$ there is some graph $S \in \mathcal{S}$ such that $f(C, S) > 0$. Let $\mathfrak{C}^{\mathcal{S}} = \bigcup_k \mathfrak{C}_k^{\mathcal{S}}$ be all contexts found in $\mathcal{S}$. For a given trivial context $C \in \mathfrak{C}_1^{\mathcal{S}}$ we let $\mathfrak{C}_{k,C}^{\mathcal{S}} \subseteq \mathfrak{C}_k^{\mathcal{S}}$ be the set of all non-trivial contexts of size k in which C can be embedded into.

If $\mathcal{S}$ is a set of sampled molecules and t_{obs} their molecular energy, we can think of $\mathfrak{C}_1^{\mathcal{S}}$ as all possible bonds found in $\mathcal{S}$. Similarly given some bond $C \in \mathfrak{C}_1^{\mathcal{S}}$, the set $\mathfrak{C}_{k,C}^{\mathcal{S}}$ represents all possible contexts found in $\mathcal{S}$ containing k bonds and originates from C.

Functions comprising contributions for contexts $\mathfrak{C}_k^{\mathcal{S}}$ up to some some order k are already an appealing class of regression models $\mathbb{F}$. They are, however, still prone to overfitting since they still involve contexts that are observed very rarely in $\mathcal{S}$.

4.1.2 *Modelling Context Influence*

For a graph $S \in \mathcal{S}$, the maximal amount of structural information is given by the contexts $C = (G, e)$ with $G \simeq_\varphi S$, i.e., by specifying S completely. In this case t_{edge} exactly determines $t_{\text{edge}}(\varphi(e))$. In practise however, this would generalize very poorly since most graphs outside the training set $\mathcal{S}$ will not match the contexts. As an alternative, therefore, we have to consider smaller contexts that are sufficiently frequent in $\mathcal{S}$ and that cover at least most of the graphs for which we wish to predict t. Not all contexts of a given size are equally frequent in $\mathcal{S}$, of course. Frequent molecular features thus may be sampled with more accuracy, i.e., larger context size, than others (not unlike longer k-mers allowing more accurate sampling of frequent motifs in alignment-free sequence comparison methods [14]). We therefore consider a nested scheme of context contributions.

To this end we introduce the map $t_{\mathfrak{C}} : \mathfrak{C}^{\mathcal{S}} \to \mathbb{R}$ that tracks how each context *contributes* to the values of t_{edge}. The values of $t_{\mathfrak{C}}$ will be defined recursively. The smallest possible contexts are the trivial ones in $\mathfrak{C}_1^{\mathcal{S}}$. Consider such a trivial context $C \in \mathfrak{C}_1^{\mathcal{S}}$. Let $\mathbb{E}_C \subseteq \mathbb{E}$ be the set of all those edges $e \in \mathbb{E}$ such that C occurs around e. The average value $t_{\mathfrak{C}}(C)$ of $t_{\text{edge}}(e)$ over all $e \in \mathbb{E}_C$ provides an approximation $t_{\text{edge}}(e)$ for every edge $e \in \mathbb{E}_C$:

$$t_{\text{edge}}(e) \approx f(C, G, e) \cdot t_{\mathfrak{C}}(C) \tag{2}$$

The accuracy of the estimate $t_{\text{edge}}(e)$ will of course be poor if $t_{\text{edge}}(e)$ is strongly influenced by its environment.

The approximation can be improved by considering contexts of size 2. Suppose that we have a given context $C \in \mathfrak{C}_1^{\mathcal{S}}$ and a context $C' \in \mathfrak{C}_{2,C}^{\mathcal{S}}$ of size 2 such that $C \sqsubseteq C'$. Moreover, let $e \in \mathbb{E}_C \cap \mathbb{E}_{C'}$, i.e., both C and C' occur around e. We can then ask if the additional structural

information given by C' influences our estimation of $t_{\text{edge}}(e)$ compared to only using $t_{\mathcal{C}}(C)$. Let $\delta_{C|C'}$ be the difference of approximating $t_{\text{edge}}(e)$ with and without including C', and let $t_{\mathcal{C}}(C')$ be the average of $\delta_{C|C'}$ over all $e \in \mathbb{E}$ where C and C' occurs around e. We can think of $t_{\mathcal{C}}(C')$ as the average contribution of any edge which contains the environment of C'. If there is no significant difference in the estimation of values in t_{edge} when including C' or not, either in terms of effect size or statistical significance over the available data in $\mathcal{S}$, we may simply discard C' and only use the original set $\mathcal{C}_1^{\mathcal{S}}$ for estimating the values of t_{edge}. On the other hand, if C' significantly influences our estimations of t_{edge}, it would suggest that we would be able to achieve a better approximation by considering the set of contexts $\mathcal{C}_1^{\mathcal{S}} \cup \{C'\}$ and setting

$$t_{\text{edge}}(e) \approx f(C, G, e) \cdot t_{\mathcal{C}}(C) + f(C', G, e) \cdot t_{\mathcal{C}}(C')$$

In this way, by determining the significance of any context in $\mathcal{C}_2^{\mathcal{S}}$ in relation to estimating the values of t_{edge}, we can obtain a set $\mathcal{K}_2 \subseteq \mathcal{C}_1^{\mathcal{S}} \cup \mathcal{C}_2^{\mathcal{S}}$ containing all contexts of size 1 and 2 that significantly influences our estimation of t_{edge}.

This idea immediately generalizes to a chain of nested sets of contexts $\mathcal{C}_1^{\mathcal{S}} = \mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq \cdots \subseteq \mathcal{K}_n$, such that $\mathcal{K}_k$ represents all significant contexts of sizes 1 to k and n is the largest number of edges of all graphs in $\mathcal{S}$. Using the same argumentation as before, a context C of size k is significant if considering $\mathcal{K}_{k-1} \cup \{C\}$ provides a better estimation of t_{edge} compared to only considering $\mathcal{K}_{k-1}$. Hence, by determining the significance of all contexts in $\mathcal{C}_k^{\mathcal{S}}$ we can construct the set $\mathcal{K}_k$, which we can use to estimate $t_{\text{edge}}(e)$ for some edge e of a given graph $G \in \mathcal{S}$:

$$t_{\text{edge}}(e) \approx \sum_{C \in \mathcal{K}_k} f(C, G, e) \cdot t_{\mathcal{C}}(C). \tag{3}$$

where $\mathcal{K}_k$ provides at least as good of an approximation as $\mathcal{K}_{k-1}$.

4.1.3 Iterative Inference of Significant Contexts

Eq. (3) provides a way to approximate $t_{\text{edge}}(e)$ edge if the significant contexts $\mathcal{K}_k$ and their contributions are known. Given a graph $S \in \mathcal{S}$ and all significant contexts $\mathcal{K}_k$ we can use Eqns. (1) and (3) to express the relation between t_{obs} and the significant contexts:

$$\begin{aligned} t_{\text{obs}}(S) &\overset{\text{Eq. (1)}}{=} \sum_{e \in E(S)} t_{\text{edge}}(e) \\ &\overset{\text{Eq. (3)}}{\approx} \sum_{e \in E(S)} \sum_{C \in \mathcal{K}_k} f(C, S, e) \cdot t_{\mathcal{C}}(C) \\ &= \sum_{C \in \mathcal{K}_k} t_{\mathcal{C}}(C) \cdot \sum_{e \in E(S)} f(C, S, e) \\ &= \sum_{C \in \mathcal{K}_k} f(C, S) \cdot t_{\mathcal{C}}(C) \end{aligned} \tag{4}$$

Introducing an error term ϵ to account for the contributions that cannot be explained by $\mathcal{K}_k$, and setting $t_{\mathcal{C}}(C) = 0$ for all $C \notin \mathcal{K}_k$ we obtain

$$\begin{aligned} t_{\text{obs}}(S) &= \sum_{C \in \mathcal{K}_k} f(C, S) \cdot t_{\mathcal{C}}(C) + \epsilon \\ t_{\text{obs}}(S) - \sum_{C \in \mathcal{K}_{k-1}} & f(C, S) \cdot t_{\mathcal{C}}(C) = \\ & \sum_{C \in \mathcal{K}_i \setminus \mathcal{K}_{k-1}} f(C, S) \cdot t_{\mathcal{C}}(C) + \epsilon \\ t_{\text{obs}}(S) - \sum_{C \in \mathcal{K}_{k-1}} & f(C, S) \cdot t_{\mathcal{C}}(C) = \\ & \sum_{C \in \mathcal{C}_k^{\mathcal{S}}} f(C, S) \cdot t_{\mathcal{C}}(C) + \epsilon \end{aligned} \tag{5}$$

Hence, if we know the values of $t_{\mathcal{C}}$ for the contexts in $\mathcal{K}_{k-1}$ we can determine the significant contexts in $\mathcal{C}_k^{\mathcal{S}}$ by solving the set of equations given by Eq. (5) while minimizing ϵ.

The decomposition of edge values $t_{\text{edge}}(e)$ into contributions of different sizes k contains undetermined degrees of freedom. Estimating the contributions in the order of increasing sizes, however, implies

$$0 \approx \sum_{S \in \mathcal{S}} \sum_{C' \in \mathcal{C}_{k,C}^{\mathcal{S}}} f(C', S) \cdot t_{\mathcal{C}}(C') \tag{6}$$

since the terms with $k > 1$ are "corrections" to the trivial edge contributions. Estimating the latter by linear regression implies that the expected value of the residual vanishes, i.e., the higher-order contribution $k > 1$ estimate contributions that average to 0 on the training set $\mathcal{S}$. The equations from Eq. (5) and (6) forms the basis of a linear equation system to find the significant contexts of size k and their contributions $t_{\mathcal{C}}$ under the assumption that we have found the significant context of size 1 to $k-1$.

We therefore consider only the subset of regression models for which all contexts are significant in $\mathcal{S}$ as our final set $\mathbb{F}$.

4.1.4 Learning Context Contributions

In what follows, we assume w.l.o.g. that the elements in $\mathcal{S}$, $\mathcal{C}_k^{\mathcal{S}}$, and $\mathcal{C}_1^{\mathcal{S}}$ are ordered and let S_i, C_i, Z_i, denote the graph at position i in $\mathcal{S}$, $\mathcal{C}_k^{\mathcal{S}}$, and $\mathcal{C}_1^{\mathcal{S}}$ respectively. We can then define a $(|\mathcal{S}| + |\mathcal{C}_1^{\mathcal{S}}|) \times |\mathcal{C}_k^{\mathcal{S}}|$ matrix $\mathbf{X}$, where each row represents the right hand side of a linear equation from Equation (5) and (6), and each column represents the coefficients for a variable $t_{\mathcal{C}}(C_j)$ in the respective linear equation. To be more precise, for every $i \in \{1, \ldots, |\mathcal{S}| + |\mathcal{C}_1^{\mathcal{S}}|\}$ and every $j \in \{1, \ldots, |\mathcal{C}_k^{\mathcal{S}}|\}$ we put:

$$\mathbf{X}_{ij} = \begin{cases} f(C_j, S_i) & \text{if } i \leq |\mathcal{S}| \\ \sum_{S \in \mathcal{S}} f(C_j, S) & \text{if } i > |\mathcal{S}|, C_j \in \mathcal{C}_{k, Z_{i-|\mathcal{S}|}}^{\mathcal{S}} \\ 0 & \text{otherwise} \end{cases}$$

Similarly we collect the left hand side values of Equation (5) and (6) in a vector $\mathbf{y}$ such that:

$$\mathbf{y}_i = \begin{cases} t_{\text{obs}}(S_i) - \sum_{C \in \mathcal{K}_{k-1}} f(C, S) \cdot t_{\mathcal{C}}(C) & \text{if } 1 \leq i \leq |\mathcal{S}| \\ 0 & \text{otherwise} \end{cases}.$$

Finally, denote by $\mathbf{t}$ the $|\mathcal{C}_k^{\mathcal{S}}| \times 1$ vector of the unknown contributions of the respective values in $t_{\mathcal{C}}$. Our task is then to solve the linear system

$$\mathbf{X} \cdot \mathbf{t} - \mathbf{y} = \epsilon \tag{7}$$

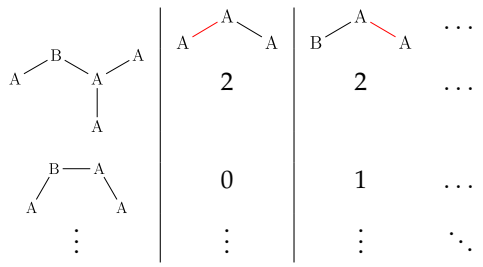


Fig. 3. Illustrated is the first two rows and columns of the matrix $\mathbf{X}$ constructed as in Sec. 4.1.4 from some sets $\mathcal{S}$ and $\mathcal{C}_2^{\mathcal{S}}$. As shown, we can think of each sample S as a row in $\mathbf{X}$, each column as come context C, and each entry as the frequency $f(C, S)$.

while collectively minimizing the error terms ϵ in a suitable fashion. Eq. (7) can of course be solved by standard linear regression methods such as ordinary least squares (OLS) [15]. However, since we have not made any assumptions on the the structure of $\mathbf{X}$ we cannot guarantee that the regressors in $\mathbf{X}$ are linearly independent. Moreover, it is not given that $|\mathcal{C}_k^{\mathcal{S}}| < |\mathcal{S}|$, and hence the matrix $\mathbf{X}$ might contain more columns than rows, further breaking the assumption of linear independence.

In addition, a method that supports feature selection will be helpful to adapt the sets $\mathcal{K}_k$ "on the fly". Obviously we would like to find a set of contexts that is as small as possible to explain the variance given by t_{obs}. Any regressor removed from the regression model using feature selection corresponds to a context removed from $\mathcal{K}_k$. A particularly convenient formulation that is able to handle linear dependence of the regressors and supports feature selection is the Least Absolute Shrinkage and Selection Operator (LASSO) [13], which corresponds to the global minimization problem

$$\min \left(\sum_{i=1}^{|\mathcal{C}_1^{\mathcal{S}}|+|\mathcal{S}|} \left(\mathbf{y}_i - \sum_{j=1}^{|\mathcal{C}_k^{\mathcal{S}}|} \mathbf{X}_{ij} \mathbf{t}_j \right)^2 + \lambda \sum_{j=1}^{|\mathcal{C}_k^{\mathcal{S}}|} |\mathbf{t}_j| \right), \quad (8)$$

where λ is some given positive scalar. Here, the term including λ acts as a L1-regularization technique to perform feature selection. As an added benefit, due to the L1-regularization term, if two regressors are collinear, the learned model tends to only include one of them by setting the other to 0. During the transformation of $\mathcal{S}$ and $\mathcal{C}_k^{\mathcal{S}}$ into a linear regression model, we only retain the structural information stored in $\mathcal{S}$ and $\mathcal{C}^{\mathcal{S}}$ in the form of the frequency counting between the two. As a consequence, if the frequencies of two contexts C_1 and C_2 in $\mathcal{C}_k^{\mathcal{S}}$ are collinear with respect to $\mathcal{S}$, we will be unable to distinguish between them during the regression step. Hence, we assume that the necessary structural information required to identify significant contexts for the whole target population is specifically stored in the frequencies between $\mathcal{C}_k^{\mathcal{S}}$ and $\mathcal{S}$, i.e., if the frequencies of C_1 and C_2 are collinear in $\mathcal{S}$ they are collinear in the target population.

4.2 Algorithmic Design

4.2.1 Algoritm

The discussions can be summarized in the algorithm CARGO, which is outlined in Alg. 1. For increasing k, it learns the significant contexts of size k using the results from previous iterations. The output of CARGO is a model t storing the significant contexts $\mathcal{K}_k$ together with their contributions $t_{\mathcal{C}}$. Using the stored information, Eq. (4) can be used to predict the values of the property of interest for any graph in the target population.

Algorithm 1 A high level view of the CARGO algorithm

```
1: function CARGO(S, t_obs)
2:     t ← the empty model S → ℝ
3:     for k ← 1 to n do
4:         X, y ← FreqMatrix(C_k^S, S, t)
5:         K, t_C ← LASSO(X, y, λ)
6:         t.append(K, t_C)
7:         if Δ(t, t_obs) ≤ thres then
8:             break
9:     return t
```

First (Line 4) the matrix $\mathbf{X}$ and the vector $\mathbf{y}$ are computed as outlined in Sec. 4.1.4. Here, the model t is used to account for the variance explained by the already found significant contexts $\mathcal{K}_{k-1}$. Next (Line 5) LASSO is used to infer the regressors from $\mathbf{X}$ and $\mathbf{y}$ as explained in Sec. 4.1.4. For any regressor included in the regression model, we store their corresponding contexts and contributions, where $\mathcal{K} \subseteq \mathcal{C}_k^{\mathcal{S}}$ contains all selected contexts of size k and $t_{\mathcal{C}}$ their corresponding context contributions. The model t is then appended with $\mathcal{K}$ along with their learned context contributions $t_{\mathcal{C}}$, such that t stores all significant contexts of size k; $\mathcal{K}_k = \mathcal{K}_{k-1} \cup \mathcal{K}$. Finally (Line 7) we check if the performance measure $\Delta(t, t_{\text{obs}})$, as stated in Prob. 1, is under some given threshold `thres`. If this is the case we return the learned model, while otherwise we learn the significant contexts of size $k+1$.

The runtime of Alg. 1 will be bounded primarily by the size of the largest context in $\mathcal{C}_k^{\mathcal{S}}$ and the size of $\mathcal{C}_k^{\mathcal{S}}$. On the other hand, the complexity of using the model to predict properties of graphs not in $\mathcal{S}$ will primarily be bounded by the number of significant contexts $\mathcal{K}_k$. From an algorithmic point of view, however, considering every context in $\mathcal{C}_k^{\mathcal{S}}$ is inefficient because the size of $\mathcal{C}_k^{\mathcal{S}}$ can scale exponentially with the size of graphs in $\mathcal{S}$. We therefore consider strategies to speedup both the learning and the prediction step by pruning the set of contexts in $\mathcal{C}_k^{\mathcal{S}}$ and $\mathcal{K}_k$ that needs to be considered.

4.2.2 Context Mining

Contexts that can be embedded in very few graphs in $\mathcal{S}$ are unlikely to contribute meaningful information to the final model because it is difficult to evaluate whether the corresponding regressor conveys a signal or not. It makes sense, therefore, to limit $\mathcal{C}_k^{\mathcal{S}}$ to include only contexts that can be embedded into a "reasonable" number of graphs in $\mathcal{S}$. In other words, we require that the corresponding regressors explains a significant variance contribution in t_{obs} of several samples in $\mathcal{S}$. To this end, we define the notion of supported contexts:

Definition 5. The *support* $\texttt{sup}(C)$ of a context $C \in \mathcal{C}^{\mathcal{S}}$ is the number of graphs in $\mathcal{S}$ which C can be embedded into. Given a positive integer τ we say that C is supported if

$\texttt{sup}(C) \geq \tau$. We denote by $\texttt{SUP}(k,\tau) \subseteq \mathcal{C}_k^{\mathcal{S}}$ the set of all supported contexts of size k.

Given a context $C = (G, e) \in \mathcal{C}^{\mathcal{S}}$ and a graph $S \in \mathcal{S}$, we have then $G \subseteq S$ if and only if C is embeddable into S. To enumerate all supported contexts, it therefore suffices to first find all graphs that are subgraph isomorphic to at least τ graphs in $\mathcal{S}$, and then to use the resulting graphs to construct all supported contexts.

The problem of enumerating all non-isomorphic subgraphs occurring in at least τ graphs is known as Frequent Subgraph Mining (FSM). It is well studied in a variety of fields of application [16]. FSM is a computationally hard problem as it involves the problem of subgraph isomorphism which is known to be NP-complete [17]. Nevertheless, several efficient algorithms exists for solving FSM.

Definition 6. Let $\mathcal{G}$ be a set of graphs and k and τ two integers such that $k > 0$ and $\tau > 0$. Then $\texttt{FSM}(\mathcal{G}, k, \tau)$ is the set of non-isomorphic graphs that contain k edges and are subgraph isomorphic to at least τ graphs in $\mathcal{G}$.

Clearly, if $\tau = 1$, then $\texttt{FSM}(\mathcal{G}, k, \tau)$ consists of all non-isomorphic subgraphs with k edges that can be found in the graphs in $\mathcal{G}$. For a given integer k, we can construct all supported contexts of size k as follows: Let $G \in \texttt{FSM}(\mathcal{S}, k, \tau)$ be a frequent graph of size k in $\mathcal{S}$. For any edge $e \in E(G)$ we can create the context $C = (G, e)$. Since $G \not\simeq G'$ for any other graph in $G' \in \texttt{FSM}(\mathcal{S}, k, \tau)$, $G \neq G'$, it follows that $C \not\simeq C'$ for any $C' = (G', e')$ where $e' \in E(G')$. It might, however, still be the case that $C \simeq C'$ for $C' = (G, e')$ where $e' \in E(G)$ due to symmetries in G. Let $\texttt{SUP}(G)$ be the set of non-isomorphic contexts created from G as explained above. Then:

$$\texttt{SUP}(k,\tau) = \bigcup_{G \in \texttt{FSM}(\mathcal{S},k,\tau)} \texttt{SUP}(G) \tag{9}$$

An example of frequent graphs and supported contexts is illustrated in Fig. 4.

Finally, denote by $\texttt{SUP}(k, \tau, m)$ the set of the m most supported contexts in $\texttt{SUP}(k, \tau)$. Instead of using all contexts of $\mathcal{C}_k^{\mathcal{S}}$ as possible significant contexts at iteration $k > 1$, as specified in Line 4 of Alg. 1, we use only $\texttt{SUP}(k, \tau, m)$. At the same time, we can accelerate the construction of $\mathbf{X}$ and $\mathbf{y}$ by noting that $C \in \texttt{SUP}(k, \tau)$ implies that its graph is a subgraph of some $S \in \mathcal{S}$, and thus its subgraphs are also contained in S. It suffices, therefore, to compute $f(C, S)$ when we previously noticed during the construction of $\texttt{SUP}(k, \tau)$ that C can be embedded into S; otherwise we immediately set $f(C, S) = 0$.

4.2.3 Context Representative

The number of contexts can be reduced further using the following observation:

Theorem 1. Given a positive integer k, let $C_1 = (G_1, e_1)$ and $C_2 = (G_2, e_2)$ be two contexts in $\mathcal{C}_k^{\mathcal{S}}$. If $G_1 \simeq G_2$ then $f(C_1, S) = c \cdot f(C_2, S)$ for all $S \in \mathcal{S}$ and some constant $c > 0$, i.e., the frequencies of C_1 and C_2 are collinear in $\mathcal{S}$.

Proof: We emphasize that $f(C_1, G_1)$ is the number of automorphisms $\varphi : V(G_1) \to V(G_1)$ that fix the edge e_1 and that $f(C_1, G_1) = f(C_1, S)$, whenever $G_1 \simeq S$. Let P_1, resp., P_2 be the set of subgraphs in S that are isomorphic to G_1, resp., G_2. Clearly, we would be able to express the frequency of C_1 in S as: $f(C_1, S) = f(C_1, G_1) \cdot |P_1|$. Similarly, $f(C_2, S) = f(C_2, G_2) \cdot |P_2|$. Since $G_1 \simeq G_2$ we have $|P_1| = |P_2|$ and thus, $f(C_1, S)/f(C_1, G_1) = f(C_2, S)/f(C_2, G_2)$. Writing this as $f(C_1, S) = \frac{f(C_1,G_1)}{f(C_2,G_2)} f(C_2, S)$ shows collinearity since $\frac{f(C_1,G_1)}{f(C_2,G_2)}$ is a constant. □

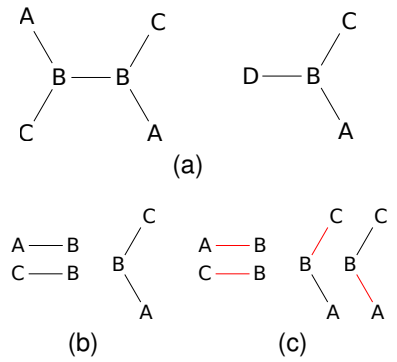


Fig. 4. Assume $\mathcal{S}$ contains the graphs illustrated in Fig. 4a and that $\tau = 2$. Then Fig. 4b illustrates the resulting subgraphs contained in $\texttt{FSM}(\mathcal{S}, 1, \tau)$ and $\texttt{FSM}(\mathcal{S}, 2, \tau)$. Similarly Fig. 4c illustrates the subgraphs contained in $\texttt{SUP}(1, \tau)$ and $\texttt{SUP}(2, \tau)$.

Thm. 1 implies that the regressors for C_1 and C_2 with $G_1 \simeq G_2$ in the matrix construction $\mathbf{X}$ are collinear in the first $|\mathcal{S}|$ rows of $\mathbf{X}$. More precisely, the regressors for C_1 and C_2 will only break collinearity in exactly the two rows of $\mathbf{X}$ corresponding to the specific Eqns. listed in (6) that includes either C_1 or C_2. This suggests that we gain an insignificant amount of information by including both C_1 and C_2 as regressors in the linear model. Therefore, we further reduce the context set by avoiding contexts with isomorphic graphs.

This simplification however breaks the equations given by Eq. (6). However, we can relax the constraints given by Eq. (6) to a single equation:

$$0 = \sum_{S \in \mathcal{S}} \sum_{C \in \mathcal{C}_1^{\mathcal{S}}} \sum_{C' \in \mathcal{C}_{k,C}^{\mathcal{S}}} f(C', S). \tag{10}$$

Clearly if the equations given by (6) holds, then (10) must hold as well. The other way does not necessarily hold, however if the relaxed version had a solution then it would also contain a solution when removing contexts as explained above, since C_1 and C_2 would be collinear in Eq. (10). We then use the relaxed Eq. (10) instead of the equations given by Equation (6) during the construction of each $\mathbf{X}$ in the algorithm given in Sec. 4.2.1. Hence, the actual set of supported contexts outlined in the previous section, $\texttt{SUP}(k, \tau)$, will only contain the set of supported contexts that are pairwisely non-isomorphic.

4.2.4 Predicting New Graphs From the Target Population

Eq. (4) can now be used to predict values of t for any graphs G in the target population, that is, the set of graphs that admit a covering by contexts (including trivial) in $\mathcal{K}_k$. To this end, it is necessary to compute the frequencies $f(C, G)$ for all $C \in \mathcal{K}_k$. As mentioned previously, such computations can be expensive. Suppose C can be embedded into C'. Then $f(C, G) = 0$ implies $f(C', G) = 0$. Hence, we would

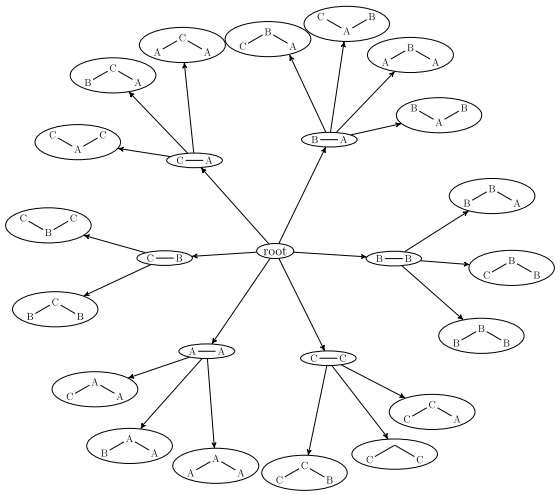
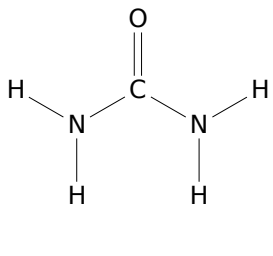


Fig. 5. Example of a context tree as defined in Definition 7



C	$f(C,S)$	$t_{\mathcal{C}}(C)$
N—C	2	6
O=C	1	-3
	4	-2
	2	7

Fig. 6. Suppose we want to estimate the molecular energy of the molecule urea (left) based on the listed significant contexts (right) and their contribution $t_{\mathcal{C}}$. Let S denote the urea molecule. Using Eq. (4) we can estimate the molecular energy $t_{\text{est}}(S)$ of S as: $t_{\text{est}}(S) = 2 \cdot 6 + 1 \cdot (-3) + 4 \cdot (-2) + 2 \cdot 7 = 15$. Note, this is just an illustrative example and in no way models the real molecular energy of urea.

only need to compute the frequency of C' if C has a non-zero frequency. This observation suggest to construct a context tree:

Definition 7. Given a set of contexts $\mathcal{K}_k$, a *context tree* is a tree $T = (\mathcal{K}_k \cup r, E)$ where r is a special vertex designated as the root and the edge set E is defined as follows: Let $C \in \mathcal{K}_k$. Then the parent of C in T is the largest context $C' \in \mathcal{K}_k$, $C \neq C'$, that is embeddable into C. If several such contexts of the same size exists a random one is chosen. If none exists C is connected to the root r.

We note that the context tree for a set of contexts is not unique. An example of a context tree is illustrated in Fig. 5. Given a graph G and equipped with a context tree T for $\mathcal{K}_k$ we can compute the frequencies of $\mathcal{K}_k$ in G with the help of a breadth first search in T starting from the root. Each time we visit a vertex C in T, except for the root, we compute the frequency of its corresponding context. If $f(C, G) > 0$, we continue down to the children of C in T. If however $f(C, G) = 0$, we can prune every descendant of C in T and set their frequencies to 0.

4.3 Parameter Selection

The runtime peformance of Alg. 1, as well as the quality of the final model t, are affected by the selection of a few key parameters. These are: The number of iterations to run (n), when contexts are considered supported (τ), the maximum number of contexts to consider from $SUP(k, \tau)$ (m), the regularization parameter of LASSO (λ), and finally when a model is considered acceptable (`thres`). The parameters n and `thres` control the size of the contexts that should be included in the model. Including contexts that are too large will make the resulting model prone to overfitting, while only considering very small context will make the model prone to underfitting. On the other hand, the parameters τ, m, and λ, controls how many contexts of a certain size should be included, and also here extreme values for any parameter might make the model prone to over/underfitting. The specific parameter values used in each experiment are listed in the relevant sections of Sec. 6.

4.4 Computational Complexity

Alg. 1 does not solve Prob. 1 exactly. Instead it tries to find a minimal set of subgraphs that can adequately be used as a model for accurate future predictions.

To implement Alg. 1 we must be able to solve the following problems: Computing $f(C, S)$, computing $\texttt{SUP}(k, \tau, m)$, and solving the LASSO objective function. Any solution to $f(C, S)$ involves the enumeration of subgraph isomorphisms between two graphs. Finding subgraph isomorphisms between graphs are known to be NP-complete [17], and moreover, there can be an exponential number of such isomorphisms. Note, this also implies that solving Prob. 1 for subgraph additive functions is NP-complete. Similarly, finding the set $\texttt{SUP}(k, \tau, m)$ involves solving FSM, which, as stated previously, is NP-complete and can again have an exponential output. LASSO can be solved in polynomial time using algorithms such as coordinate descent [18]. The complexity of such algorithms, however, depends on the number of regressors, which in turn depends on the size of $\texttt{SUP}(k, \tau, m)$.

As we shall see in Sec. 6, all of the above problems can be solved efficiently in practice. On one hand, this is due to our ability to adjust peformance using the parameters m and τ that control the number of contexts. On the other hand, for real-world datasets, we can often obtain good models with very small contexts, which trivializes both the subgraph isomorphism or subgraph enumeration problem.

5 Implementation

The algorithm described in Sec. 4.2.1 was implemented as a python library, with the computational extensive tasks being implemented in `C++`. CARGO is available from https://imada.sdu.dk/Employees/daniel/cargo/.

The software relies on the solution of several well-known computational problems, in particular subgraph isomorphism enumeration, FSM, and LASSO regression. Computing the frequency of a context in a graph efficiently is integral for both the model construction and prediction of new graphs, and ultimately depends on enumerating subgraph isomorphisms. For this purpose we employ the vf2 algorithm [19] implemented in the boost graph library [20]. The enumeration of all frequent subgraphs of $\mathcal{S}$ relies on a multi-threaded implementation of the `gSpan` algorithm [21]

in the C++ library gBolt [22]. Note, for this work, we have modified the algorithm such that in addition to accepting a set of graphs as input it also accepts a positive integer k and outputs a set of non-isomorphic frequent subgraphs that all contains exactly k edges. The libraries scikit-learn [23] and numpy [24] supplied the implementation of the LASSO regression. For reading and visualizing graphs we use the software package MØD [4].

For all data-sets tested in Sec. 6 CARGO ran in less than a minute on a standard desktop PC, however the runtime will of course depend on the exact parameters chosen, e.g. the maximum number of iterations to be run during model construction. The prediction of values on new graphs, when a model has been learned, as explained in Sec. 4.2.4 all took less than a second to estimate values of the whole data-set.

6 Results

6.1 Evaluation of CARGO

Although this work was motivated by the need to predict Gibbs free energies for a very wide range of molecules to guide large scale explorations of chemical spaces with MØD, we investigate the general applicability and efficiency of our approach for several different application scenarios. Thus the problems addressed below were chosen to generalize at least one aspect of the typically very focused applications of group contribution methods.

We start with an artificially constructed example so that the performance of CARGO can be assessed relative to an absolute ground truth. We then show that CARGO performs very well on the most typical task, the prediction of thermodynamic properties, (e.g.., boiling points of hydrocarbones). Finally, we apply CARGO as a predictor in a scenario where no functional groups as used in classical group contribution methods based on metabolic compounds exist. This is illustrated based on prediction of energies which are based on expensive quantum chemical computation of a training set.

Throughout, the standard error of prediction $\mathrm{SE} = \sqrt{(1/\mathcal{S})\sum_{S\in\mathcal{S}}(t_{\mathrm{obs}}(S) - t_{\mathrm{est}}(S))^2}$ is used to quantify the quality of the trained models. Here, t_{est} denotes the property values estimated with CARGO. In order to determine the λ-values in the LASSO regression, Eq. (8), 10-fold cross-validation was used. The accuracy of CARGO on graphs not in $\mathcal{S}$ was assessed using double cross-validation with 10-fold resampling also in the outer validation. The resulting standard error is denoted by $\mathrm{SE}_{\mathrm{cross}}$. For brevity in the following CARGO will refer to results obtained for the whole data-set, while $\mathrm{CARGO}_{\mathrm{cross}}$ will refer to cross-validation, respectively. Similarly, the number of significant contexts found by CARGO denoted by $\mathcal{K}$ and $\mathcal{K}_{\mathrm{cross}}$ for whole data-sets and for the cross-validated data-sets respectively. We ran CARGO 10 times on each data-set to assess the stability of CARGO and report the standard deviations of these replicates.

The residuals of the regression are expected to approximately follow a normal distribution if CARGO correctly capture a linear relation between the contributions of the significant contexts and the observed property values. We therefore compute the percentage of residuals R_1 and R_2 that falls within one and two standard deviations respectively. For normally distributed residuals we expect $R_1 \approx 68\%$ and $R_2 \approx 95\%$.

6.2 Artificial Test Data: Colored Trees

We consider 3-colored trees and the property t_{obs} as an additive model. To define $\mathcal{S}$, we randomly generated 100 trees each with 12, 15, 18, and 20 vertices and assigned colors randomly to the vertices. In order to define t_{obs}, we manually assigned values $(10, -20, 30, -40, 50, -60)$ to the 6 possible "base" contributions for the coloring of the vertices of a single edge and then selected 15 contexts of size 2, 15 contexts of of size 3, and 10 contexts of size 4 randomly from a collection of contexts that all occur in at least 10% of the samples in $\mathcal{S}$. We then assigned values to them picked from a uniform distribution on the intervals $[1, 20]$, $[1, 10]$, and $[1, 5]$ and choose random sign $+$ or $-$ to approximate Eq. (6) for each edge. The decreasing upper bound of the intervals was chosen to reflect the intuition that higher order context yield corrections to lower-order ones. Eqns. (1) and (3) were then used to compute $t_{\mathrm{obs}}(S)$ for all $S \in \mathcal{S}$. An example of the step-wise construction of the synthetic data-set is illustrated in Fig. 7. Running CARGO and $\mathrm{CARGO}_{\mathrm{cross}}$ we obtained

SE	$\mathrm{SE}_{\mathrm{cross}}$	$\mathcal{K}$	$\mathcal{K}_{\mathrm{cross}}$	R_1	R_2
7.6 ± 0.3	9.9 ± 0.1	106.3 ± 15.3	101 ± 4.6	69%	95%

A single run of CARGO and $\mathrm{CARGO}_{\mathrm{cross}}$ was generated in $6m33s$ respectively. The parameters used was $n = 4$, $\tau = 40$, $m = \infty$. The residuals indeed approximate normal distributions, both for CARGO and $\mathrm{CARGO}_{\mathrm{cross}}$, as seen from Fig. 8a and the values of R_1 and R_2.

Moreover, we see that SE and $\mathrm{SE}_{\mathrm{cross}}$ are very similar as well as $\mathcal{K}$ and $\mathcal{K}_{\mathrm{cross}}$ indicating some amount of stability even when removing 10% of the data-set. The robustness of the model is further confirmed by the low standard deviation across runs. The accuracy is illustrated in Fig. 8b, where we see the cross-validated estimates plotted against the observed values for a single run. The estimated edge contributions (trivial contexts)

	base cont.	est. cont
A–A	-40.0	-37.8
A–B	30.0	25.7
A–C	10.0	11.0
B–B	-60.0	-62.6
B–C	-20.0	-15.9
C–C	50.0	42.5

are obtained in the 0th iteration of CARGO and already closely resemble their *a priori* values. We do not expect a perfect match since the higher order contributions in our artificial model were chosen in a way that does not strictly enforce Eq. (6). The trivial contexts thus compensate for the average contribution of the higher-order terms. The variance of the energy values in each trivial context when embedded into a sample is shown in Fig. 8d.

In the next iterations, larger contexts stepwisely correct for the errors incurred by using only trivial contexts. Fig. 8c shows that the standard error of CARGO drops rapidly by introducing contexts of size 2 and then 3. Not much is gained thereafter, however. Note, this is also where much of the variance of $\mathcal{K}$ comes from, as some runs chose very few contexts of size 4. It seems that the variance given by the true significant contexts of size 4 was instead explained by

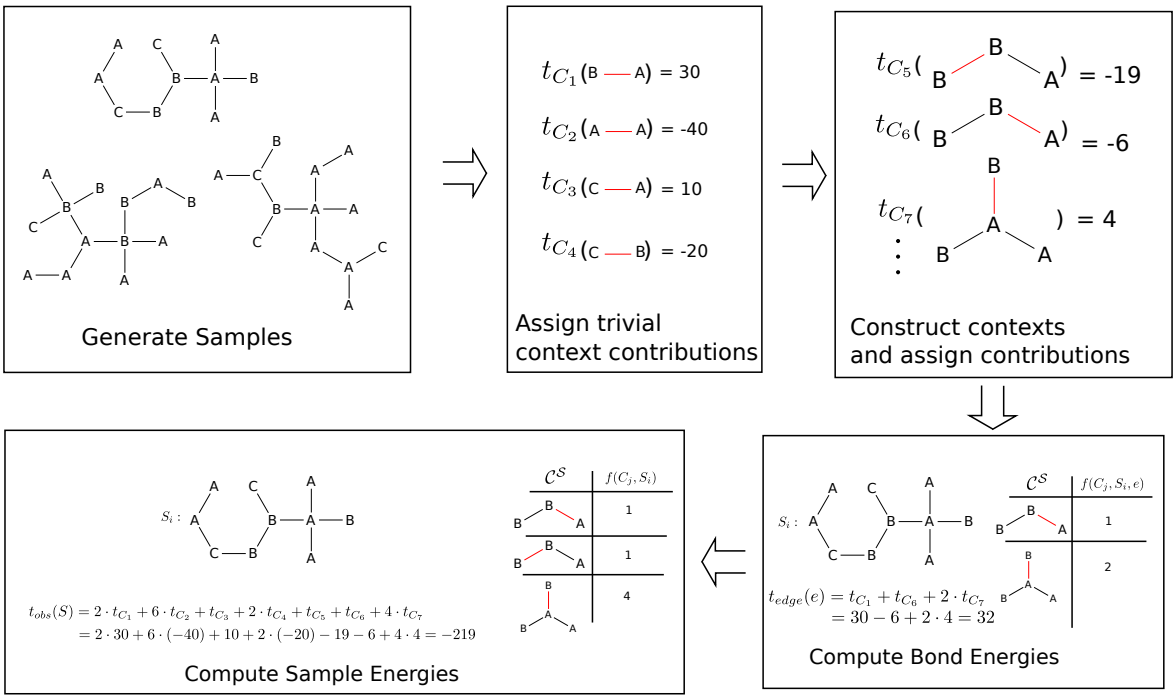


Fig. 7. Illustrative example of the pipeline involved in creating the synthetic dataset. First, the samples are generated as random trees. Bond types from the constructed samples are identified and assigned energies. Next, frequent contexts are mined from the samples and assigned energies simulating how bonds are biased by contexts. Finally, the energy of each bond in a sample is computed, and the total energy of a sample is computed based on the bond energies. For easy reference t_{C_i} refers to the contribution $t_{\mathfrak{C}}(C_i)$ of the illustrated context C_i.

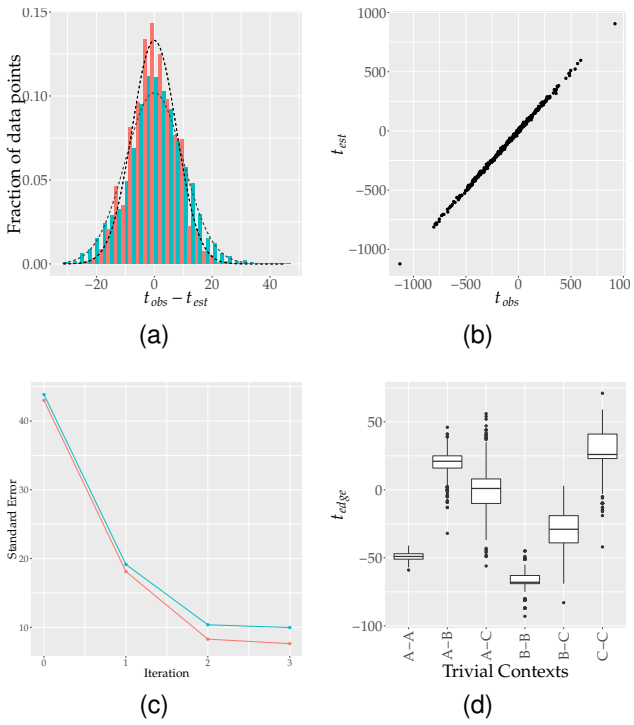


Fig. 8. Illustrated results for Sec. 6.2. *Top-left:* The distribution of residuals for CARGO (red bars) and CARGO$_{\text{cross}}$ (blue bars). The black and gray curve depicts a normal distribution with the same mean and standard deviation as the two depicted distributions. *Top-right:* t_{obs} vs. t_{est}, for a single run of CARGO$_{\text{cross}}$. *Bottom-left:* The mean standard error of CARGO (red) and CARGO$_{\text{cross}}$ (blue) at each iteration. *Bottom-right:* The variance in the true values of t_{edge} grouped by their corresponding trivial contexts.

additional contexts of size 2 and 3, even though they were not present during the construction of the data-set. This is not too surprising, since larger contexts introduce more linear dependence: Clearly if a context C can be embedded into a context C', we would expect some amount of linear dependence between them. The variance of contexts of size 4 being explained by contexts of size 2 and 3 is further supported by the fact that $\mathcal{K}$ and $\mathcal{K}_{\text{cross}}$ both contained around 100 contexts on average, which is around three times as many contexts as was used for the construction of the data-set. The number of contexts found using CARGO could of course be reduced by increasing the λ parameter of LASSO, however, the additional contexts seem to produce a more accurate model even for CARGO$_{\text{cross}}$.

Finally, in Fig. 8c we see a small increasing gap between SE and SE$_{\text{cross}}$ at increasing iterations. This is explained by the fact that the inclusion of more and larger contexts allows for more degrees of freedom during the learning of the model, which in turn leads to more potential for overfitting.

In summary, we have shown that the CARGO approach is indeed applicable to data-sets containing contexts of relative small size, occurring in at least 10% of the samples, and which decrease its contribution with expanding size. In such an environment, we gain excellent accuracy and robustness in the model and we observe a clear benefit of including larger significant context in later iterations.

6.3 Thermodynamics in Metabolic Networks

As a first real-life application we test the quality of the CARGO models on a dataset of Gibbs Free Energy of molecules appearing in metabolic networks provided by [25]. The dataset is comprised of 221 compounds with

thermodynamic values that have been determined experimentally.

We removed all compounds with less than 4 atoms as well as compounds containing unusual bond types present in less than 3 graphs in the original dataset. After preprocessing the original dataset shrunk to 196 entries (88%). The reason for removing very small compounds is that they tend to have special structures that do not generalize. For example a context $\mathrm{H-O-H}$ appears only in water, $\mathrm{H_2O}$, and thus, if included as a regressor, would ensure a perfect prediction of the value of water. On the other hand, if we insist on a minimal number of occurrences of C in the training data, $\mathrm{H-O-H}$ is excluded and it is impossible to train such cases accurately. A viable strategy is to first exclude such cases, as we have done here, and if desired reinclude them in a final training designed to incorporate such special cases into the model without compromising the part of the model that generalizes to large classes of molecules.

Fig. 9a shows the number of samples containing a given atom type for all atom types included in the dataset. Specifically, we see that the atoms types Br, F, I, and S only occur in 3 samples. This is problematic since sulphur, for instance, is known to play a significant role in the total energy of a compound, however the low number of samples makes it impossible for any approach to learn and validate such information. Moreover, specifically for sulphur the corresponding variables for the bond types $\mathrm{S=C}$ and $\mathrm{S-C}$ are collinear, and hence indistinguishable which is not the case in reality.

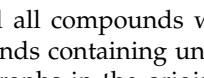
Instead of determining the λ-value for LASSO by cross-validation, we used here a fixed parameter $\lambda = 0.1$ to save computational resources and this choice was experimentally found to produce good results in practice. Moreover, due to the scarcity of some atom types such as sulfur we chose to run CARGO$_{\text{cross}}$ as a *"leave-one-out"* approach to ensure at least two of these atom types are in $\mathcal{S}$ during training. For a single run, the models CARGO and CARGO$_{\text{cross}}$ were generated in $4s$ and $5m55s$ respectively. The parameters used was $n = 4$, $\tau = 3$, $m = 400$. An overview of the results is tabulated below:

SE	SE$_{\text{cross}}$	$\mathcal{K}$	$\mathcal{K}_{\text{cross}}$	R_1	R_2
6.4 ± 0.0	15.1 ± 2.2	185 ± 10.1	185 ± 6.39	74%	94%

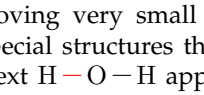
We see that the residuals also here approximate normal distributions both for CARGO and CARGO$_{\text{cross}}$, as seen from Fig. 9b and the values of R_1 and R_2. The average absolute observed value of a sample in $\mathcal{S}$ was found to be 460, the standard error thus is just above 1%, showing the CARGO is quite accurate. This conclusion is further corroborated by the scatter plot in Fig. 9c, which compared the estimated t_{est} and observed t_{obs} values for CARGO$_{\text{cross}}$.

Note, that a large part of the uncertainty in SE$_{\text{cross}}$ is concentrated in few large outliers. The worst of such offenders was the molecule Sulfite SO_3^{-2} which in some runs of CARGO$_{\text{cross}}$ was mispredicted with a residual of 247, much larger than any misprediction otherwise observed. Indeed, SE$_{\text{cross}} = 13.4 \pm 0.0$ when ignoring Sulfite during the evaluation of SE$_{\text{cross}}$.

Still, we observe a notable discrepancy between SE and SE$_{\text{cross}}$ indicating some amount of overfitting. However, this is not too unexpected since $\mathcal{S}$ contains very few samples with certain atom types. Namely, the worst outliers across runs of CARGO$_{\text{cross}}$ were the molecules Sulfite SO_3^{-2}, Sulfate SO_4^{-2}, and Carbonic acid H_2CO_3. Of these both Sulfite and Sulfate contains sulfur which is likely to explain the poor prediction. Carbonic acid also is an "exotic" compound



as the context $\mathrm{O-C-O}$ barely appears elsewhere and hence cannot be learned in our setting. At the same time, it is not well represented by either diol/diesther $\mathrm{O-C-O}$



carboxylic acid $\mathrm{C-O}$ sub-contexts, whence we are left with the large residual.

Another source of instability in the model seems to stem from choosing contexts that occurs in too few samples. This makes the model prone to overfitting by "hiding" variance in the rare regressors. Running CARGO such that only contexts that occurs in at least 25% of all samples were chosen, resulted in the standard errors of SE $= 17.2$ and SE$_{\text{cross}} = 19.7$ when ignoring the outliers mentioned above. The distribution of residuals is also illustrated in Fig. 9e, where we see the distribution of residuals of CARGO$_{\text{cross}}$ much closer resembling the distribution of residuals from CARGO. While the overall accuracy of the model decreases we see an increase of robustness in the model. With this in mind, context mining as explained in Sec. 4.2.2 should be seen as a tradeoff between accuracy and overfitting of the trained model.

As with the synthetic dataset, including increasingly larger contexts improves the accuracy of the model both for CARGO and CARGO$_{\text{cross}}$ confirming the introduction of contexts improves the model in real data-sets as well. Note that, even though SE decreases when including contexts of size 4, no notable improvement is seen in SE$_{\text{cross}}$.

Finally, comparing CARGO to the model obtained in [25], we see that they achieved standard errors of SE $= 1.9$ and SE$_{\text{cross}} = 2.2$ from a set of 85 manually curated groups. In this respect, CARGO performs markedly worse. It should however be noted, that the comparison is not completely fair, since in [25] the Gibbs Free Energy change of reactions was used in addition to the energy of molecules. As a result, their sample size consisted of 869 molecules and reactions used for training compared to the 196 samples used here. Second, it should be emphasized that our approach is not intended to outperform methods tailored to a specific chemistry but instead to achieve good accuracy in a wide variety of application scenarios.

6.4 Boiling Point

For testing the quality of the CARGO model when applied to the prediction of normal boiling points we used a dataset provided by [26]. It consists of 186 acyclic molecules with hetero atoms, including acyclic ether, peroxides and sulfur analogues, whose normal boiling points have been measured.

The models CARGO and CARGO$_{\text{cross}}$ were generated in $17s$ and $2m35s$ respectively. The parameters used was $n = 4$, $\tau = 3$, $m = 200$. An overview of the results is listed below:

SE	SE$_{\text{cross}}$	$\mathcal{K}$	$\mathcal{K}_{\text{cross}}$	R_1	R_2
5.6 ± 0.2	6.6 ± 0.1	55.7 ± 10.1	57.1 ± 3.8	81%	95%

Again, we see that R_1 and R_2 together with Fig. 11a confirms the assumption that the residuals of CARGO and CARGO$_{\text{cross}}$ approximates a normal distribution.

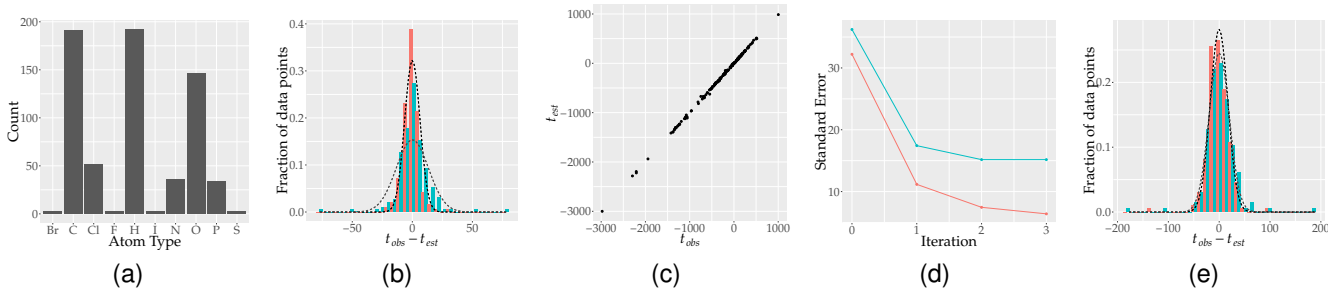


Fig. 9. Illustrated results for Sec. 6.3. *Top-left:* Distribution of atoms found in each sample in $\mathcal{S}$. *Top-right:* The distribution of residuals for CARGO (red bars) and CARGO$_{\text{cross}}$ (blue bars). The black and gray curve depicts a normal distribution with the same mean and standard deviation as the two depicted distributions. *Top-right* t_{obs} vs. t_{est}, for a single run of CARGO$_{\text{cross}}$. *Middle-left:* The mean standard error of CARGO (red) and CARGO$_{\text{cross}}$ (blue) at each iteration. *Middle-right:* The variance in the true values of t_{edge} grouped by their corresponding trivial contexts. *Bottom:* Distribution of residuals for CARGO (red bars) and CARGO$_{\text{cross}}$ (blue bars) when only considering contexts that occur in 25% of all samples in $\mathcal{S}$. The black and gray curve depicts a normal distribution with the same mean and standard deviation as the two depicted distributions.

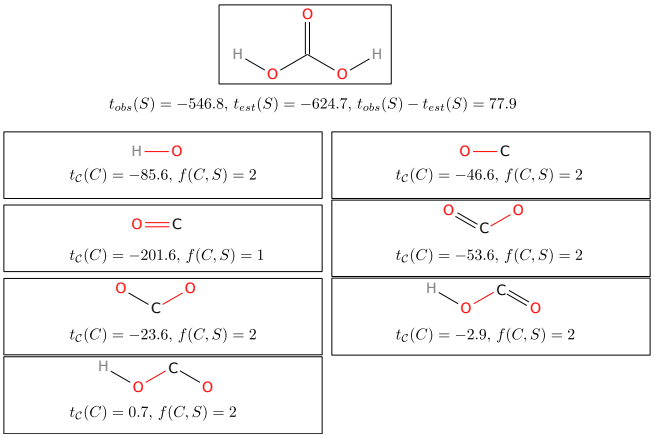


Fig. 10. Example of the prediction of carbonic acid, S, for one run of CARGO$_{\text{cross}}$, along with the significant contexts used to compute $t_{\text{est}}(S)$.

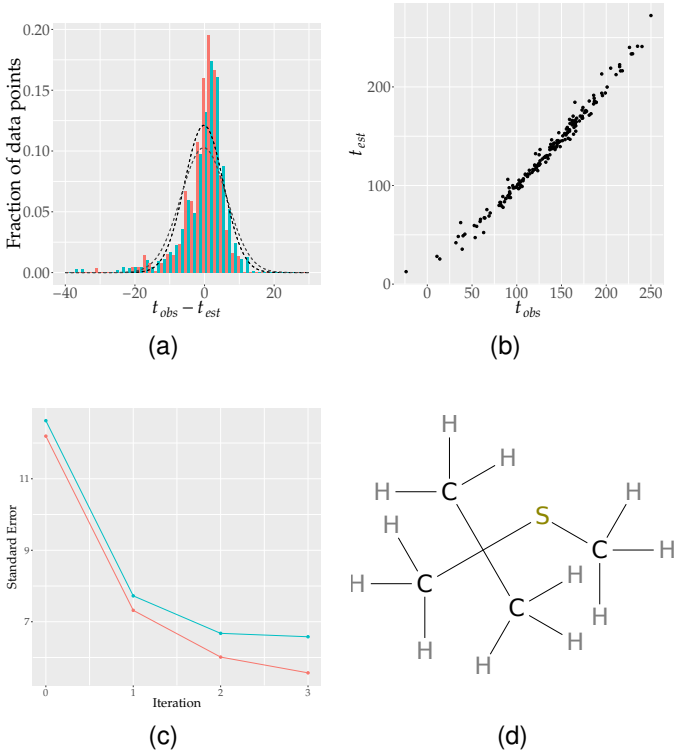


Fig. 11. Illustrated results for Sec. 6.4. *Top-left:* The distribution of residuals for CARGO (red bars) and CARGO$_{\text{cross}}$ (blue bars). The black and gray curve depicts a normal distribution with the same mean and standard deviation as the two depicted distributions. *Top-right:* t_{obs} vs. t_{est}, for a single run of CARGO$_{\text{cross}}$. *Bottom-left:* The mean standard error of CARGO (red) and CARGO$_{\text{cross}}$ (blue) at each iteration. *Bottom-right:* An example of the sample **CC(C)(C)SC** in $\mathcal{S}$ with $t_{\text{obs}}(S) = 101.5$, while CARGO$_{\text{cross}}$ obtained the estimation $t_{\text{est}}(S) = 101.54$.

It should be noted, however, that the overlaid normal distributions seem to be slightly shifted to the left. This indicates that CARGO seem to have a tendency to over-predict the values of certain samples in $\mathcal{S}$. This is made even clearer by examining Fig. 11b, plotting t_{obs} against t_{est} for a single run of CARGO$_{\text{cross}}$, where we see a slight upwards bend in the plotted residuals. More specifically, it seems like CARGO will tend to over-predict samples with the highest or lowest observed values.

In general however, the accuracy of CARGO seems to be good when considering the average absolute value of a sample in $\mathcal{S}$ was 133, making the standard error just above 4%. Moreover, we see that SE and SE$_{\text{cross}}$ are very similar, indicating robustness in predictive capabilities of CARGO, which is further reinforced by the low standard deviation across runs.

The largest context identified by CARGO was of size 4. Also here we see that $\mathcal{K}$ and $\mathcal{K}_{\text{cross}}$ are very similar, however the standard deviation for $\mathcal{K}$ is also fairly high. This might indicate some variance in the number of contexts selected, however as we have seen it does not influence the accuracy of CARGO too heavily.

Regression methods using graph kernels have been used successfully for the prediction of boiling point values in a variety of data-sets [27], [28]. In [26] several methods of this type were benchmarked using 10-fold cross-validation. The best performance was reported for [28] with a standard error of 6.75, which is slightly worse than the standard error of 6.6 achieved by CARGO. Since both the data and the evaluation protocol is identical to the one we used for CARGO, the performance can be compared directly.

6.5 Thermodynamics in Sugar Chemistry

In contrast to the metabolite data of section 6.3 the 149 (multisets of) sugar compounds in [29] are chemically very homogenous, comprising a carbon backbone equipped with hydrogen and oxygen atoms.

Where in previous sections each sample in $\mathcal{S}$ specifically referred to a single connected graph, a sample in this dataset refers to a collection of molecules, (called "flasks" in [29]), and effectively translates into a graph with multiple connected components (i.e., several sugar compounds). Moreover, the number and types of atoms is constant across samples, ie. each sample contains the same number of vertices connected via edges into a different multi-component graph. While the approach of CARGO as described in this paper assumes each sample point is a connected graph representing a single molecule, it is trivial to extend CARGO by simply counting the frequency of contexts in each molecule included in the sample.

A single run of CARGO and CARGO$_{\text{cross}}$ was generated in 20s and 3m10s respectively. The parameters used was $n = 5$, $\tau = 3$, $m = 300$. An overview of the results is tabulated below:

SE	SE$_{\text{cross}}$	$\mathcal{K}$	$\mathcal{K}_{\text{cross}}$	R_1	R_2
0.08 ± 0.0	0.15 ± 0.0	89.7 ± 11.9	87.0 ± 2.9	71%	94%

Like in the previous sections, the expected percentages of residuals to fall within one and two standard deviations coincides with our results. Moreover, we see that the residuals for CARGO and CARGO$_{\text{cross}}$ both approximate a normal distribution, which can clearly be seen in Fig. 12a.

The average absolute observed value for a sample in $\mathcal{S}$ was 0.9, the standard error is hence around 10%, which seems fairly accurate. This is also emphasized in Fig. 12a where we plot t_{obs} against t_{est} for a single run of CARGO$_{\text{cross}}$, although some spread in the residuals is observed.

Notably, a few clear outliers are present in Fig. 12b. Upon examination of $\mathcal{S}$, we see that these compounds are the sample of the collection of molecules `OC1C(C1O)O`, `C=O` and the molecule `OCOC1C(C1O)O`. The larger molecules in both samples contain a cyclopropane ring. In this 3-member ring, the angles between succesive C — C bonds are strongly distorted from thermodynamically favorable 109.5° to only 60° degrees, which introduces high ring strain. Furthermore, both the formaldehyde $H_2C=O$ molecule in the first sample and the cyclopropane substructure contained in both samples are very special. As a consequence, the issue of identifying the relevant contexts therefore the follows same arguments as for water H_2O in Sec. 6.3 hold. Upon further investigation in the dataset, it was found that these two samples are the only ones with this specific cycle, suggesting such cycles might be important contexts for prediction but not identified by CARGO$_{\text{cross}}$. This hypothesis is further reinforced by the compounds not being outliers when including the whole training set for fitting using CARGO.

The significant context found for both CARGO and CARGO$_{\text{cross}}$ was of size 5. Moreover, the fact that $\mathcal{K}$ found almost the same number of contexts as $\mathcal{K}_{\text{cross}}$, indicates robustness in the selection of significant contexts. Note however, the fairly high standard deviation of $\mathcal{K}$ across runs, that seems to be caused by CARGO sometimes including too many contexts of size 5 that does not improve the accuracy. From Fig. 12c we see that the largest improvements in accuracy was gained by including contexts up to size 4. The inclusion of larger contexts still improved the accuracy of CARGOand CARGO$_{\text{cross}}$, however much less dramatically.

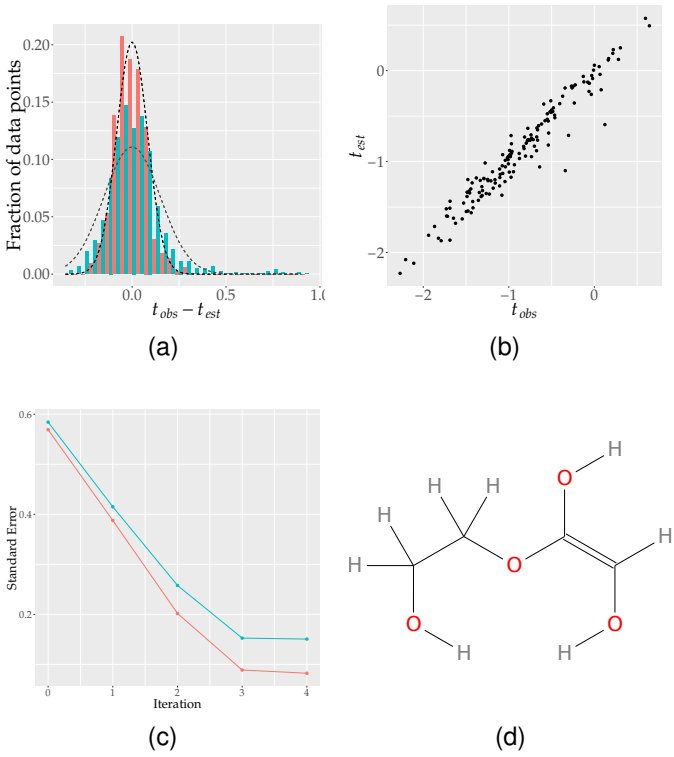


Fig. 12. Illustrated results for Sec. 6.5. *Top-left:* The distribution of residuals for CARGO (red bars) and CARGO$_{\text{cross}}$ (blue bars). The black and gray curve depicts a normal distribution with the same mean and standard deviation as the two depicted distributions. *Top-right:* t_{obs} vs. t_{est}, for a single run of CARGO$_{\text{cross}}$. *Bottom-left:* The mean standard error of CARGO (red) and CARGO$_{\text{cross}}$ (blue) at each iteration. *Bottom-right:* An example of the sample **OCCOC(=CO)O** in $\mathcal{S}$ with $t_{\text{obs}}(S) = -0.827$, while CARGO$_{\text{cross}}$ obtained the estimation $t_{\text{est}}(S) = -0.83$.

7 Concluding Remarks

We have presented here a general framework for (additive) group contribution methods that does not require a prescribed lists of "groups". Instead, we defined a generic notion of context via local subgraphs in such a manner that the learning step not only determines the regression coefficient but also the relevant context graphs. To this end we employ LASSO regression, which implicitly performs feature selection. The approach described here is centered around a reference edge around which the contexts are localized in G. This structure is motivated by the molecular energies, which are at least conceptually explained by chemical bonds. This is also true for many other thermodynamic properties of molecules, which are also determined by chemical bonds and their local interactions. We have shown that the implementation in software package CARGO is robust and applicable to a wide variety of application scenarios.

A conceptual advantage of the CARGO model is that step-wise definition of contributions with increasing context size k makes the models relatively easy to interpret. Starting from bond contribution, the larger and larger contexts enter as corrections to smaller, more generic contributions. As a

consequence, it becomes easy to estimate the contributions of chemical substructures of interest, even if they do not appear as regressors/contexts themselves. This structure of the model also makes it possible to account for small, exceptional molecules that do not fit well into the generic regression model. Examples such as water, carbon dioxide, carbonic acid, or formaldehyde contain contexts that are essentially private to these molecules. Adding those as additional regressors enforces that the measures valued t_{obs} for such exceptional cases without the need for extra rules and separate tables.

In the current model, we assume additive contributions of the regressors. Suitable transformations can be used to accomodate other functional dependencies. For example, kinetic constants depend exponentially on (activation) energies, hence we expect that a log-transform of the data will be helpful. It is also possible to use other "anchors" than reference edges. Similar to many graph-kernel methods, for example, on could just as well use vertices and vertex-centered contexts [30]. The example of RNA secondary structure shows that the choice of a suitable basis structure is important for group contribution methods. In this case, individual edges are poor predictors, while the isometric cycles of G provide an excellent approximation. Extensions of the energy model of RNA secondary structure go beyond cycles and include also partially overlapping pairs of cycles as regressors [31].

Acknowledgments

This work is supported in by Novo Nordisk Foundation grant NNF19OC0057834 and by the Independent Research Fund Denmark, Natural Sciences, grant DFF-7014-00041.

References

[1] K. G. Joback and R. C. Reid, "Estimation of pure-component properties from group-contributions," *Chem. Eng. Commun.*, vol. 57, pp. 233–243, 1987.

[2] A. Fredenslund, R. L. Jones, and J. M. Prausnitz, "Group-contribution estimation of activity coefficients in nonideal liquid mixtures," *AIChE J.*, vol. 21, pp. 1086–1099, 1975.

[3] Z. Kolská, M. Zábranský, and A. Randova, "Group contribution methods for estimation of selected physico-chemical properties of organic compounds," in *Thermodynamics-Fundamentals and Its Application in Science*, R. Morales-Rodriguez, Ed. IntechOpen, 2012, pp. 135–161.

[4] J. L. Andersen, C. Flamm, D. Merkle, and P. F. Stadler, "A software package for chemically inspired graph transformation," in *Graph Transformation, ICGT 2016*, ser. Lecture Notes Comp. Sci., R. Echahed and M. Minas, Eds., vol. 9761. Berlin, Heidelberg, D: Springer Verlag, 2016, pp. 73–88.

[5] ——, "Chemical Transformation Motifs — Modelling Pathways as Integer Hyperflows," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8171738

[6] D. Dor and M. Tarsi, "Graph decomposition is NP-complete: A complete proof of Holyer's conjecture," *SIAM Journal on Computing*, vol. 26, no. 4, pp. 1166–1187, 1997.

[7] Y.-R. Luo, *Comprehensive handbook of chemical bond energies*. CRC press, 2007.

[8] M. S. Silberberg, *Principles of general chemistry*. McGraw-Hill Higher Education New York, 2007.

[9] J. Marrero and R. Gani, "Group-contribution based estimation of pure component properties," *Fluid Phase Equilibria*, vol. 183, pp. 183–208, 2001.

[10] Z. Kolská, J. Kukal, M. Zábranský, and V. Røužička, "Estimation of the heat capacity of organic liquids as a function of temperature by a three-level group contribution method," *Ind. Eng. Chemistry Res.*, vol. 47, pp. 2075–2085, 2008.

[11] E. Noor, H. S. Haraldsdóttir, R. Milo, and R. M. T. Fleming, "Consistent estimation of Gibbs energy using component contributions," *PLoS Comp. Biol.*, vol. 9, p. e1003098, 2013.

[12] D. H. Mathews, M. D. Disney, J. L. Childs, S. J. Schroeder, M. Zuker, and D. H. Turner, "Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure," *Proc. Natl. Acad. Sci. USA*, vol. 101, no. 19, pp. 7287–7292, 2004.

[13] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. Royal Statistical Soc. Ser. B*, vol. 58, pp. 267–288, 1996.

[14] S. Vinga and J. Almeida, "Alignment-free sequence comparisona review," *Bioinformatics*, vol. 19, no. 4, pp. 513–523, 2003.

[15] L. S. Aiken, S. G. West, and S. C. Pitts, "Multiple linear regression," in *Handbook of psychology*. Wiley Online Library, 2003, pp. 481–507.

[16] C. Jiang, F. Coenen, and M. Zito, "A survey of frequent subgraph mining algorithms," *Knowledge Engineering Rev.*, vol. 28, pp. 75–105, 2013.

[17] S. A. Cook, "The complexity of theorem-proving procedures," in *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, ser. STOC '71. New York, NY, USA: ACM, 1971, pp. 151–158.

[18] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of statistical software*, vol. 33, no. 1, p. 1, 2010.

[19] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento, "A (sub) graph isomorphism algorithm for matching large graphs," *IEEE Trans. Pattern Anal. Machine Intelligence*, vol. 26, pp. 1367–1372, 2004.

[20] B. Schäling, *The boost C++ libraries*. Boris Schäling, 2011.

[21] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," in *Proceedings of the IEEE International Conference on Data Mining*, 2002, pp. 721–724.

[22] K. Zhou, "gBolt–very fast implementation for gSpan algorithm in data mining," https://github.com/Jokeren/DataMining-gSpan, 2017.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Machine Learning Res.*, vol. 12, pp. 2825–2830, 2011.

[24] S. van der Walt, S. C. Colbert, and G. Varoquaux, "The NumPy array: a structure for efficient numerical computation," *Computing in Science & Engineering*, vol. 13, pp. 22–30, 2011.

[25] M. D. Jankowski, C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis, "Group contribution method for thermodynamic analysis of complex metabolic networks," *Biophysical journal*, vol. 95, pp. 1487–1499, 2008.

[26] "Greyc's chemistry dataset," https://brunl01.users.greyc.fr/CHEMISTRY/#Acyclic, accessed: 2018-29-10.

[27] S. Liu, C. Cao, and Z. Li, "Approach to estimation and prediction for normal boiling point (nbp) of alkanes based on a novel molecular distance-edge (mde) vector, λ," *J. Chem. Inf. Computer Sci.*, vol. 38, pp. 387–394, 1998.

[28] B. Gaüzere, L. Brun, and D. Villemin, "Two new graphs kernels in chemoinformatics," *Pattern Recognition Letters*, vol. 33, pp. 2038–2047, 2012.

[29] D. Rappoport, C. J. Galvin, D. Y. Zubarev, and A. Aspuru-Guzik, "Complex chemical reaction networks from heuristics-aided quantum chemistry," *J Chem Theory Computation*, vol. 10, pp. 897–907, 2014.

[30] F. Costa and K. D. Grave, "Fast neighborhood subgraph pairwise distance kernel," in *Proceedings of the 26th International Conference on Machine Learning*. Haifa: Omnipress, 2010, pp. 255–262.

[31] C. Höner zu Siederdissen, S. H. Berhart, P. F. Stadler, and I. L. Hofacker, "A folding algorithm for extended RNA secondary structures," *Bioinformatics*, vol. 27, pp. i129–i137, 2011.

[32] D. H. Turner and D. H. Mathews, "NNDB: The nearest neighbor parameter database for predicting stability of nucleic acid secondary structure," *Nucleic Acids Res.*, vol. 38, pp. D280–D282, 2009.

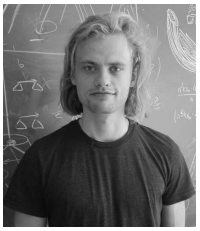
[33] R. Lorenz, S. H. Bernhart, C. Hoener zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker, "ViennaRNA package 2.0," *Alg. Mol. Biol.*, vol. 6, p. 26, 2011.

[34] C. B. Do, D. A. Woods, and S. Batzoglou, "CONTRAfold: RNA secondary structure prediction without physics-based models," *Bioinformatics*, vol. 22, pp. e90–e98, 2006.

Christoph Flamm received his Doctorate in Chemistry from the University of Vienna in 1998. He was conferred the *venia docendi* in 2006 from the same school and is since then Associate Professor at the Institute for Theoretical Chemistry. His research focuses on questions at the border between Chemistry and Computer Science.

Marc Hellmuth received the Diploma degree in Economathematics (2007) and a PhD degree in Computer Science (2010) from U. Leipzig. Following PostDoc positions in Leipzig and Saarbrücken he was Juniorprofessor for Biomathematics at U. Greifswald (2015-2020) and received a *venia legendi* from Saarland University in 2016. He works as a lecturer at U. Leeds. His research focuses on the mathematical understanding of biological and chemical structures

Daniel Merkle received the Diploma degree in Computer Science (1997) and the PhD degree in Applied Computer Science (2002) from U. Karlsruhe. He worked as Assistant Professor with the Dept. of Computer Science, Leipzig, until 2008, then as Associate Professor and since 2017 as Full Professor at the Department of Mathematics and Computer Science, U. Southern Denmark. His research interests include combinatorial and algorithmic approaches for Chemistry.



Nikolai Nøjgaard Received his Masters degree in Computer Science from the University of Southern Denmark in 2018. Currently, he is studying for his PhD. at the same school and at the University of Greifswald. His research area include algorithmic approaches for life sciences.

Peter F. Stadler received his PhD in Chemistry from U. Vienna in 1990, where he then worked as Assistant and Associate Professor for Theoretical Chemistry. Since 2002 he is Full Professor for Bioinformatics at U. Leipzig. He is External Professor at the Santa Fe Institute, External Scientific Member of the Max Planck Society, Corresponding Member Abroad of the Austrian Academy of Sciences, and Honorary Professor of the Universidad Nacional de Colombia.