Department of Computer Science
Algorithmic Bioinformatics

Universitätsstr. 1 D–40225 Düsseldorf

Assignment of partial charges to identical atom neighborhoods of drug-like molecules

Richard Michael Geller

Bachelor Thesis

Submission:	31.07.2019
Supervisor:	Prof. Dr. Gunnar Klau
Second Assessor:	Dr. Frank Gurski

Declaration

I hereby confirm that this thesis is my own work. I have documented all sources and tools used. Any direct or indirect quote has been marked as such clearly with specification of the source.

Düsseldorf, July 31, 2019

Richard Michael Geller

Abstract

Here we present an approach to assign identical partial charges to atoms with chemically equivalent environments. We show how atoms of a molecule can be grouped into sets of atoms with equivalent environments by comparing the atom neighborhoods with subgraph isomorphism and graph canonization. As an extension to the charge assignment problem, we introduce the symmetric charge assignment problem where the task is to select a charge for every atom of a molecule from a range of candidate charges. Charges have to be selected such that chemically equivalent atoms receive the same charge and such that the sum of partial charges matches the total known charge of the molecule with a slight margin for error. Each candidate charge has a score that is based on the observed frequency of that charge in a database of already parameterized molecules. By selecting charges that maximize the sum of scores under these conditions we assign charges that are frequently assigned in similar chemical environments. We show that the symmetric charge assignment problem can be modeled as a multiple-choice knapsack problem variant and show that our variant remains weakly NP-complete. We give an Integer Linear Programming formulation and design a transformation algorithm to assign identical partial charges using an existing pseudo-polynomial time Dynamic Programming algorithm. We evaluate our method on a database of about 261.000 parameterized molecules and compare our computed charges to quantum-mechanically (QM) derived charges. We show that our method results in very similar average charge errors, but increases the usability of the computed charges in molecular dynamics simulations.

Contents

1 Introduction

Molecular dynamics (MD) simulations are computational simulations of molecules and intermolecular interactions and play a key role in modern analysis of (bio)-chemical systems. They simulate motion on a fixed time-scale as the result of forces acting between atoms in large systems of molecules and are used to gain insight into molecular processes like ligand binding pathways, protein folding and many more.

Force fields describe molecules and are parameterized according to the molecular structure of a molecule and the properties of a system. Partial charges in these force fields describe the electron density on all atoms throughout a molecule. They are float values associated with each atom's center and model electrostatic interactions. On small molecules (with less than 40 atoms) partial charges can be fitted with an electrostatic potential obtained from quantum mechanics (QM) calculations [1], but with larger molecules, computational costs increase significantly [2].

As noted by Malde et al. [1], partial charge assignment is a major challenge in the development of force fields, as it is not possible to relate partial charges to physical observable values. Partial charges are just an approximation to the electron distribution and methods to assign these partial charges are based on assumptions made to the underlying physics.

This paragraph is based on the introduction given in (Engler et al., [2]). The local environment of an atom is described by the surrounding atoms and their bonds in close proximity to the atom. The local environment influences the partial charge of an atom heavily which opens up the possibility to estimate partial charges of an unparameterized molecule based on charges assigned in equivalent local environments of other molecules. Equivalent local environments have the same local pattern of connected atoms and thus similar chemical properties. Therefore, we can assign charges to atoms of a query molecule by referring to the charges assigned in already parameterized molecules that contain environments equivalent to the query molecule atom environments. Since partial charges are just an approximation, the charges assigned to equivalent environments in different reference molecules can vary. If we screen a database of already parameterized molecules for charges assigned in equivalent atom environments, we end up with a range of possible charges for every atom of the query molecule. Standard approaches to assigning charges are to then simply take average of the candidate charges as the assigned charge, but these approaches often fail to match the total charge of a molecule.

The charge assignment problem as formulated by Engler et al. [2] is to select a charge from a range of candidate charges for every atom of the query molecule such that the sum of charges is close to the known total integer charge, and such that the frequency, with which the charges occur in equivalent environments of the reference molecules, is maximized. Selecting charges that maximize the score leads to charges frequently assigned in equivalent environments.

ϵ-MCKP is a multiple-choice-knapsack based approach by Engler et al. [2] to solve the charge assignment problem and, whilst charges selected with ϵ-MCKP are on average comparable to the reference charges [2], ϵ-MCKP sometimes assigns inconsistent charges, with atoms of a molecule that have equivalent environments receiving different charges.

Partial charges with differing values on equivalent chemical groups are inappropriate for molecular dynamics simulations [1]. A molecule with these differing partial charges could show different properties of otherwise identical functional groups, for example, ligands might only interact with one side of a protein that actually has two identical binding sites, reducing the predicted protein's activity by 50%. This results in inaccurate force field descriptions of the molecules.

1.1 Objective

In this bachelor thesis we aim to improve the consistency, reliability and symmetry of partial atomic charge assignment within drug-like molecules by introducing the *symmetric charge assignment problem* which, in addition to the charge assignment problem, requires assigned charges to be identical on atoms with chemically equivalent environments. To solve the symmetric charge assignment problem, we introduce a generalization of ϵ-MCKP, called ϵ-EMCKP and then show how ϵ-EMCKP can be solved with Integer-Linear-Programming and Dynamic Programming. We explain how identical atom neighborhoods can be identified and grouped into equivalence sets to use in conjunction with ϵ-EMCKP for assignment of identical partial charges.

We evaluate the accuracy of ϵ-EMCKP with a leave-one-out analysis in which we simulated the assignment of charges on new molecules not included in a database. Our data set (reference and validation data) is a snapshot of the Automated Topology Builder (ATB) and Repository [1] that contains roughly 261.000 molecules. ATB charges are calculated on the density functional theory (DFT) level of quantum mechanics with an electrostatic potential fitting approach (ESP) based method. Average charge errors (compared to the ATB) of our method are almost identical to those of ϵ-MCKP whilst providing more consistent and reliable charges on molecules that contain atoms with equivalent environments.

1.2 Preliminaries

Like in (Engler et al., [2]), molecules are defined as molecular graphs $G = (V, E, t)$ where vertices $v \in V$ are atoms, edges $\{u, v\} \in E$ are bonds between atoms u and v and a function $t : V \to \Sigma$ labels atoms by their atom type (alphabet Σ). Bonds between atoms are usually described in chemistry by an additional parameter that indicate the number of electrons used in that bond (single, double and triple bonds) but Engler et al.

[2] did not include the bond type, as (Martin Engler, personal communication, July 23, 2019) bond-types would be inferred by a set of arbitrary rules and their chemical experts were of the opinion that bond types had a neglect-able influence on partial charges.

As atom types we use the IACM type code of the GROMOS force field. This type code gets assigned by the ATB. Instead of providing a single label for every chemical element, there are multiple different labels for the chemical elements, e.g. oxygen has five different atom types and carbon has eight different atom types [1]. These IACM values are inferred from the connectivity, chemical environment and sometimes net charge of the atom [1]. Therefore, IACM atom types provide a more detailed and specific description of the local atom environment than the chemical elements.

2 ϵ-MCKP

Engler et al. [2] describe the problem of assigning partial charges to atoms of a query molecule from a set of candidate charges with a multiple choice knapsack approach. Every atom i of a molecule is mapped to a class N_i that contains items j with weights $w_{i,j}$ corresponding to candidate partial charges and profits $p_{i,j}$ corresponding to the charge's score. This score depends on the frequency with which the candidate charge is observed in equivalent atom neighborhoods of other molecules. The known total charge of the query molecule is mapped to the capacity c and the sum of assigned partial charges has to equal this total charge. The error ϵ limits the maximal allowed difference between the sum of assigned charges and the known total charge.

Engler et al. [2] assign the charges by selecting one item from each class such that these selected items maximize the sum of profits (the score of a solution). They subject the sum of selected item's weights to be in the range $[c-\epsilon, c+\epsilon]$ which, in combination with maximization of profits, yields an optimal solution to ϵ-MCKP and thus to the charge assignment problem. The weight $w_{i,j}$ of the selected item j in class N_i represents the charge that they assign to the atom i.

Definition 2.1 (ϵ-MCKP, decision version)**.** Given a variable $K \geq 0$, capacity $-\infty < c < \infty$, error $\epsilon \geq 0$, m classes $N_1, \ldots, N_m$ of items $j \in N_i$ with profit $p_{i,j} \geq 0$ and weight $-\infty < w_{i,j} < \infty$, select exactly one item from each class, such that the sum of weights of the selected items is in the range $[c-\epsilon, c+\epsilon]$ and the sum of profits of the selected items is equal or larger than K.

2.1 Solving ϵ-MCKP with Dynamic Programming

Engler et al. [2] have provided two algorithmic ways of solving ϵ-MCKP, one with Integer Linear Programming and one that is an adaption to the pseudo-polynomial Dynamic

Programming (DP) MCKP algorithm.

The DP algorithm requires weights, capacity and error to be positive and integer, since they dictate the size of the DP-table P and are used in the step-wise bottom-up creation of the DP-table. Multiplying weights and error with an appropriate factor results in integer values. For every class N_i, integer weights $w_{i,j}$ are then converted to non-negative integer weights $\tilde{w_{i,j}}$ that are defined by subtraction of the weight by the minimum weight of their class. The capacity $\tilde{c}$ is defined by subtracting the original capacity with the minimum weights of all classes.

The size of P is $m \times (\tilde{c} + \tilde{\epsilon})$ where m is the total number of classes, $\tilde{c}$ is the converted capacity and $\tilde{\epsilon}$ is the converted error. The converted values are used for the remainder of the DP algorithm.

The field $P[k, d]$ of the DP-table contains the highest profit and optimal solution that can be achieved with a sum of charges that equals exactly d with a solution using exactly one item from each of the first k classes. $P[k, d]$ is calculated as the maximum profit over all items j of class k we get by adding the profit of item j to the profit that was achieved by an optimal solution for capacity $d - w_{k,j}$ with the first $k - 1$ classes.

$P[k, d]$ has the value $-\infty$ if there is no solution to reach d by using exactly one item from each class of the first k classes. The starting point of the recursion is $P[0, 0] = 0$ and all other fields are $-\infty$.

$$P[0, d] = \begin{cases} 0 & \text{if } d = 0 \\ -\infty & \text{else} \end{cases}$$

$$P[k, d] = \max \begin{cases} P[k-1, d - w_{k,j}] + p_{k,j} & \text{for } j \in N_k \text{ and } d - w_{k,j} \geq 0 \\ -\infty \end{cases}$$

Since an optimal solution is allowed to have a capacity in the range $[\max(\tilde{c} - \tilde{\epsilon}, 0), \tilde{c} + \tilde{\epsilon}]$, an optimal solution can be found by backtracking which items were used to build the maximum profit p^* with

$$p^* = \max\{P[m, d] : \max\{\tilde{c} - \tilde{\epsilon}, 0\} \leq d \leq \tilde{c} + \tilde{\epsilon}\}$$

If p^* is $-\infty$, then there exists no solution to ϵ-MCKP, since no combination where we take one converted item from each class equals a sum of weights in the range $[\max(\tilde{c} - \tilde{\epsilon}, 0), \tilde{c} + \tilde{\epsilon}]$. This means that there is no combination of items with one item from each class whose original unconverted sum of charges is in the range $[c - \epsilon, c + \epsilon]$.

2.2 Limitation of ϵ-MCKP

Every atom has a set of possible charges (class N_i) and each of these charges has a profit. The purpose of a standard multiple choice knapsack (MCKP) problem is to identify items that maximize the profit whilst not exceeding the capacity [see [3]] with selection of one item from each class. Often the consequence of this is selecting items that do not have the highest profit of their class, because the trivial solution of only selecting the best items in each class would not fit the given capacity.

ϵ-MCKP alters the standard MCKP and constraints the sum of weights to a range $[c - \epsilon, c + \epsilon]$, with ϵ being a user-defined fixed allowed error from the total charge. Sometimes those charges with the highest profits/frequencies, i.e. the ones most often occurring in equivalent sub-molecules, still can not be selected on every atom. Then in some (or all) classes items have to be selected that do not have the highest profit.

In some cases, dependent on the molecule and all candidate charges and profits, this may lead to different selected items on atoms that have equivalent environments. This is not ideal, as atoms with equivalent environments within a molecule should have the same charge assigned to them, since that environment is the only partial charge determining factor in ϵ-MCKP.

This issue is of particular importance when looking at molecules with reoccurring fragments like some proteins consisting of identical repeated subsections. Here we would expect atoms in repeating identical structures to have a charge that is consistent (identical) across all repeated sections. With ϵ-MCKP, there are no constraints to the charges other than the total sum condition.

The consequence of this can be seen when assigning partial charges to Benzene (molID 342920), see Fig. 1. According to the structure of Benzene, we would expect all carbon atoms and all hydrogen atoms to have the same charge as they all have identical local environments. With ϵ-MCKP, the C1-atom gets a charge assigned that is different from the other partial charges of the carbon atoms C2 to C6, whereas we would expect the charge to not differ at all from the other carbon charges. As expected, the hydrogen atoms are all assigned the same charge.

Overall, these wrongly assigned charges that appear sometimes, make the charges assigned with ϵ-MCKP unreliable, as this behavior can not be predicted beforehand and is random. In this Benzene molecule example, any of the other carbon atoms could have received the different charge instead of the C1-atom, as they are all co-optimal solutions to the one displayed.

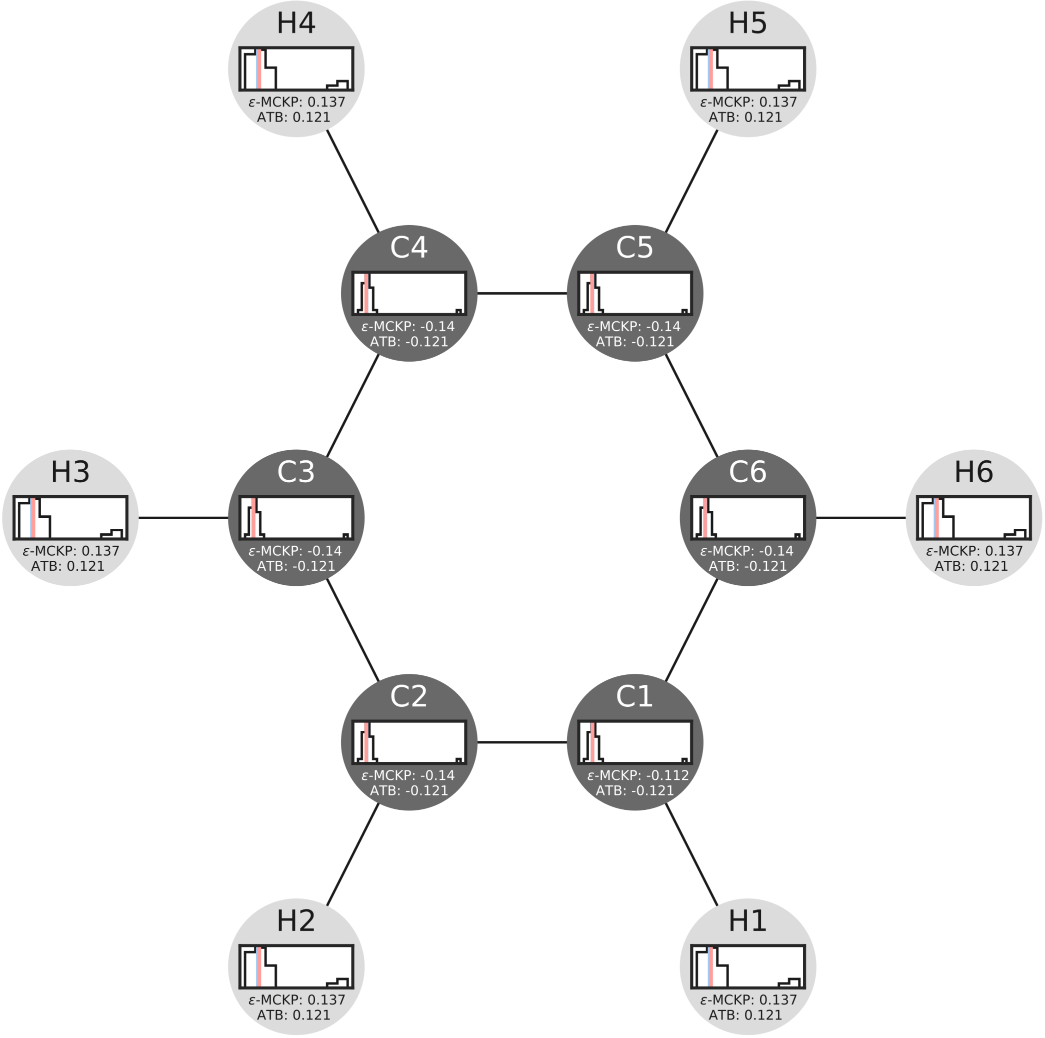

Figure 1: Benzene (molID 342920) charges assigned with ϵ-MCKP. The data set used was the one used in the leave-one-out analysis. Double bonds are not included. Note that the C1 atom gets a different charge than the C2 to C6 carbon atoms

To change this, we only allow solutions that assign the same charge to all atoms with equivalent environments. We supply additional information about which atoms should get the same charge assigned to them, according to the similarity of their local environments. We describe how those atoms get identified, grouped together to equivalence sets and how ϵ-MCKP needs to be modified to process this additional information and assign charges accordingly.

3 Atom neighborhoods

Like Engler et al. [2], we represent atom environments by k-neighborhood induced subgraphs. The k-neighborhood of an atom is the set that consists of all atoms that are at most k bonds away from the atom. The k-neighborhood induced subgraph includes the k-neighborhood and all edges between the vertices of the k-neighborhood.

We define a k-neighborhood of a node like in (Engler et al., [4]) and (with a slightly different notation) like in (Engler et al., [2]).

Definition 3.1 (k-neighborhood)**.** The k-neighborhood of a node $u \in V$ is defined recursively as the set of nodes, for which a path of length $\leq k$ exists:

$$N^k(u) = \begin{cases} \{u\}, & \text{if } k = 0 \\ N^{k-1}(u) \cup \{\{w | (v, w) \in E, v \in N^{K-1}(u)\}\} & \text{if } k \geq 1 \end{cases}$$

3.1 Comparing local atomic environments

To determine whether atoms have the same chemical environment, we compare the structural properties of their neighborhood-induced subgraphs with subgraph isomorphism.

Vertices (atoms) $v \in V$ in molecular graphs are colored with a function $t : V \to \Sigma$ to atom types Σ. Molecular (sub)-graphs are isomorphic if they are structurally identical and the node colors are also identical. This can be defined by the existence of a bijection which transforms one (sub)-graph into the other whilst keeping the graph structure (edge relations) and vertex coloring the same.

Definition 3.2 (Isomorphism of molecular graphs)**.** $G = (V, E, t)$ is isomorphic to $G' = (V', E', t')$ if there exists a bijective function $f : V \to V'$ such that

$$\forall u \in V : t(u) = t'(f(u))$$

$$\forall u, v \in V : \{u, v\} \in E \Leftrightarrow \{f(u), f(v)\} \in E'$$

We call the k-neighborhoods $N^k(u)$ and $N^k(v)$ of two different nodes (atoms) $u, v \in V$ of a molecular graph *identical* if the k-neighborhood induced subgraphs $G[N^k(u)]$ and $G[N^k(v)]$ are isomorphic to each other. This represents atoms u and v having equivalent chemical environments.

We sometimes refer to atoms of a molecule with identical k-neighborhoods as *symmetric* in the rest of this thesis. Note that this does not imply the existence of a symmetry axis on which the whole molecule or the atom neighborhood is mirrored, this is just another term for describing that two atoms have identical k-neighborhoods and therefore should get identical charges.

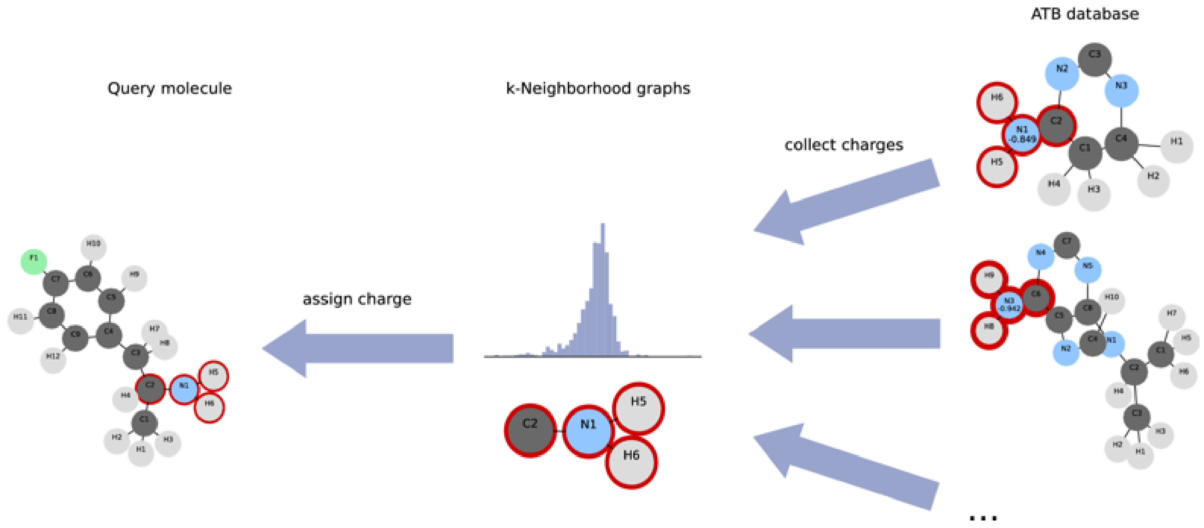

Figure 2: Overview of partial charge assignment mechanism in ϵ-MCKP. For each atom, charges get collected from molecules with isomorphic k-neighborhoods, then get condensed to a histogram from which an optimal solution is selected. Source: [2]

3.2 Isomorphism classes

In an isomorphism class all members are isomorphic to each other.

To assign charges with ϵ-MCKP, Engler et al. [2] only use the molecule structure to collect reference charges from atoms with equivalent neighborhoods in other molecules and not afterwards. In that collection step, they determine the membership of the atom neighborhood induced subgraph to an isomorphism class for every atom of the query molecule. Then they can collect the charges from already parameterized atoms whose neighborhood induced subgraphs are members of the same isomorphism class to create a range of possible charges for every atom of the query molecule (see Fig. 2).

3.3 Symmetric charge assignment problem

In the symmetric charge assignment problem, the task is to assign partial charges to all atoms of a molecule from a set of candidate charges such that the sum of all partial charges equals a known total charge with an allowed margin for error, and to assign charges such that all chemically equivalent atoms receive the same charge.

We group all atoms of a molecule into equivalence sets, where all atoms of an equivalence set have identical neighborhoods.

3.4 Classification into equivalence sets with canonical keys

A canonical form is a function that assigns to each labeled graph an isomorphic labeled graph that is a unique representative of its isomorphism class [see [5]]. A canonical key is the string representation of that unique representative.

As noted in (Engler et al., [2]), to collect the candidate charges, we match the k-neighborhood induced subgraph $G[N_k(v)]$ of every atom v of the query molecule against the repository. If there is no match, we try to match the $k-1$-neighborhood induced subgraph $G[N_{k-1}(v)]$ again and we do this iteratively until k equals zero. The canonical key of the isomorphism class of the first neighborhood-induced subgraph $G[N_{k'}(v)]$ with $k \geq k' \geq 0$ for which a matching isomorphism class is found in the repository, is taken to represent the neighborhood of atom v. For this isomorphism class of $G[N_{k'}(v)]$ the charges in the repository are collected and the canonical key serves as an identifier to the atom neighborhood induced subgraph isomorphism class.

To create the equivalence sets, we compare the canonical keys of the successfully matched k'-induced neighborhood subgraphs of all atoms. If atoms have the same canonical key then they have equivalent k'-neighborhoods (as they belong to the same isomorphism class) and we place them into the same equivalence sets.

Due to the canonical keys being required to collect the reference charges, we only add the computational effort it takes to split the atoms into equivalence sets where all atoms have identical keys. We classify the atoms by using a dictionary/hash-map and mapping the atoms by their canonical key. If a collision of keys during mapping happens, then these atoms have identical k'-neighborhoods and are put into the same equivalence set.

3.4.1 Limitations with different sized neighborhoods

A limitation with different sized neighborhoods in a molecule is that atoms with different k-neighborhoods but identical smaller k'-neighborhoods could be grouped into the same equivalence set. This might happen if on two different atoms $u, v \in V$ the subgraphs $G[N_k(u)]$ and $G[N_k(v)]$ have no matches within the repository but the subgraphs $G[N_{k'}(u)]$ and $G[N_{k'}(v)]$ with $k' < k$ are successfully matched and are identical. Therefore they yield the same canonical key and the same charge distributions. The results is: the k-neighborhoods are different but the atoms are put into the same equivalence set.

We explain this behavior by assuming that the k-neighborhoods $G[N_k(u)]$ and $G[N_k(v)]$ are not representative enough of the atom's local environment, because they have no matches in other molecules. Instead we think that the k'-neighborhoods are more suited for representation since they occur in other molecules. For us, this justifies grouping the atoms together even though their k-neighborhoods are different.

4 ϵ-EMCKP Formulation

We need to modify the ϵ-MCKP formulation to include the additional information on equivalence sets. In this section we create an extended model of ϵ-MCKP which includes these equivalence sets to assign identical charges.

As a generalization of ϵ-MCKP, we define the ϵ-Equivalence-Multiple-Choice-Knapsack-Problem (ϵ-EMCKP) which adds the equivalence sets $E_1, \dots, E_k$ to the ϵ-MCKP formulation. Like ϵ-MCKP is similar to the charge assignment problem [2], ϵ-EMCKP is similar to the symmetric charge assignment problem. We map atoms i to classes N_i with weights $w_{i,j}$ and profits $p_{i,j}$ and the total charge of a molecule to capacity c. Chemically equivalent atoms are grouped into equivalence sets $\{E_1, \dots, E_k\}$. In a feasible ϵ-EMCKP solution, all classes in an equivalence set have to have the same item selected in order to assign identical charges to equivalent atoms.

A prerequisite for every equivalence set E_l is that all classes N_i with $i \in E_l$ are required to have element-wise identical weights and profits. This is needed because to select identical items, all atoms of an equivalence set need to have the same select-able items in the first place. We call two items of different classes the same if their weights and profits are identical.

Definition 4.1 (ϵ-EMCKP, decision version)**.** Given

- a variable $K \geq 0$,
- capacity $-\infty < c < \infty$,
- error $\epsilon \geq 0$,
- m classes $N_1, \dots, N_m$ of items $j \in N_i$ with profit $p_{i,j} \geq 0$ and weight $-\infty < w_{i,j} < \infty$,
- k disjoint non-empty equivalence sets $E_1, \dots, E_k$ of items $i \in \{1, \dots, m\}$ with the identifier i of each class N_i being in exactly one equivalence set and all classes in an equivalence set having the same items,

select exactly one item from each class N_i such that the sum of weights of the selected items is in the range $[c - \epsilon, c + \epsilon]$ and such that the sum of profits of the selected items is equal or larger than K and such that for each equivalence set $E_l \in \{E_1, \dots, E_k\}$ the same item j is selected in both classes N_i and $N_{i'}$ if $i, i' \in E_l$.

Note that if two classes N_i and N_j have identical weights and profits this does not imply that they have to be in the same equivalence set E_l.

4.1 Complexity

Since ϵ-EMCKP is a generalization of ϵ-MCKP, we can deduce that it is weakly NP-complete as well. This is shown in the formal proof below:

Theorem 4.2. (ϵ-EMCKP) is weakly NP-complete.

Proof. We first have to show that ϵ-EMCKP is in NP:
On a given ϵ-EMCKP instance we can verify a solution S by checking whether S would be a valid solution to the ϵ-MCKP problem you obtain by omitting sets $\{E_1, \ldots, E_k\}$. This verifies that only a single item has been selected from each class, that the sum of selected weights is in the range $[c - \epsilon, c + \epsilon]$ and that the sum of selected profits is larger than K. Since ϵ-MCKP is weakly NP-complete [2], this verification process can be done in polynomial time.

We also have to verify if the items contained in S comply with the equivalence sets. This means that for every equivalence set $E_l \in E_1, \ldots, E_k$ the selected items in S that belong to classes which are grouped together in this equivalence set E_l, are the same. This can be verified in polynomial time as well by checking for each equivalence set whether there is any difference in the selected items.

We show that ϵ-EMCKP is weakly NP-hard with a polynomial-time many-one reduction: ϵ-MCKP $\leq_p$ ϵ-EMCKP.
Let I be a given ϵ-MCKP instance with classes $N_1, \ldots, N_m$ and let I' be the ϵ-EMCKP instance we get by adding an equivalence set $E_i = \{i\}$ for every class N_i to I. We create an equivalence set E_i for every class N_i and therefore bypass the equivalence set constraint as every class only has to have identical selected items to itself, which is always true. By using this polynomial time transformation, instances I and I' become equivalent:

$$S \text{ is a solution to } I \Leftrightarrow S \text{ is a solution to } I'$$

□

Like Engler et al. [2], to solve the symmetric charge assignment problem we are not interested in the decision-version of ϵ-EMCKP problem but rather in the optimization version which can be obtained in a similar fashion to ϵ-MCKP by "omitting [...] variable K and maximizing the sum of profits" (Engler et al., [2]).

In the remainder of this thesis we will look at the optimization version of ϵ-MCKP and ϵ-EMCKP.

5 Solving the symmetric charge assignment problem

Solving the symmetric charge assignment problem with ϵ-EMCKP instead of solving the charge assignment problem with ϵ-MCKP defines which atoms have to get identical charges, as all atoms of an equivalence set have to get the same charge. If there are multiple co-optimal solutions that satisfy the symmetric charges condition, then like in ϵ-MCKP, one solution out of the co-optimal solutions may be freely chosen.

5.1 ILP

A straight forward way of solving ϵ-EMCKP is formulating the problem as an integer linear problem and then solving it with commonly known solvers.

Engler et al. [2] have already given an ILP formulation of ϵ-MCKP:

$$\begin{aligned}
\text{maximize} \sum_{i=1}^{m} \sum_{j \in N_i} x_{i,j} p_{i,j} & & \text{(1a)}\\
\text{subject to} \sum_{i=1}^{m} \sum_{j \in N_i} x_{i,j} w_{i,j} \geq c - \epsilon & & \text{(1b)}\\
\sum_{i=1}^{m} \sum_{j \in N_i} x_{i,j} w_{i,j} \leq c + \epsilon & & \text{(1c)}\\
\sum_{j \in N_i} x_{i,j} = 1 & \quad \text{for } 1 \leq i \leq m & \text{(1d)}\\
x_{i,j} \in \{0, 1\} & \quad \text{for } 1 \leq i \leq m,\ j \in N_i & \text{(1e)}
\end{aligned}$$

$x_{i,j}$ is a binary variable that indicates whether item j in class N_i is selected ($x_{i,j} = 1$) or not ($x_{i,j} = 0$). Constraint (1d) restricts selection of items to exactly one item for every class and constraints (1b) and (1c) require the total sum of selected weights to be in the range $[c - \epsilon, c + \epsilon]$. Maximization of the profit of selected items (1a) under these conditions results in an optimal solution to ϵ-MCKP.

We adapt the ILP formulation of Engler et al. [2] by adding additional constraints for each equivalence set $E_l \in \{E_1, \ldots, E_k\}$.

$$x_{i,j} = x_{i',j} \quad \text{for } E_l \in \{E_1, \ldots, E_k\},\ i = E_{l_1},\ i' \in E_{l_{>1}},\ j \in N_i \tag{2}$$

The effect of adding constraint (2) is selection of item j in class N_i requires selection of item j in class $N_{i'}$ and vice versa. i can be fixed to the first class of an equivalence set ($i = E_{l_1}$) as all classes in an equivalence set have equal weights and profits and their order is not of importance.

If we add constraint 2 to the ϵ-MCKP ILP then we get an ϵ-EMCKP ILP, as we add the equivalence sets in the same manner as we first introduce them as an extension to ϵ-MCKP in Chapter 4.

This results in the ϵ-EMCKP ILP:

$$
\begin{aligned}
\text{maximize} \sum_{i=1}^{m} \sum_{j \in N_i} & x_{i,j} p_{i,j} && && \text{(3a)} \\
\text{subject to} \sum_{i=1}^{m} \sum_{j \in N_i} & x_{i,j} w_{i,j} \geq c - \epsilon && && \text{(3b)} \\
\sum_{i=1}^{m} \sum_{j \in N_i} & x_{i,j} w_{i,j} \leq c + \epsilon && && \text{(3c)} \\
\sum_{j \in N_i} & x_{i,j} = 1 && \text{for } 1 \leq i \leq m && \text{(3d)} \\
& x_{i,j} = x_{i',j} && \text{for } E_l \in \{E_1, \ldots, E_k\}, && \text{(3e)} \\
& && i = E_{l_1},\ i' \in E_{l_{>1}},\ j \in N_i && \\
& x_{i,j} \in \{0, 1\} && \text{for } 1 \leq i \leq m,\ j \in N_i && \text{(3f)}
\end{aligned}
$$

5.2 Transformation to ϵ-MCKP

In this section we introduce another approach to determining an optimal solution for ϵ-EMCKP.

The key idea is to transform an ϵ-EMCKP instance into an ϵ-MCKP instance, solve that transformed instance and then transform the solution of that transformed instance back to a solution of the ϵ-EMCKP instance.

In Chapter 4.1 we gave a polynomial-time many-one reduction of the decision versions of ϵ-MCKP to ϵ-EMCKP and in this Chapter we describe a polynomial-time many-one reduction of ϵ-EMCKP (optimization variant) to ϵ-MCKP (optimization variant), showing that we can solve ϵ-EMCKP optimally by solving the transformed ϵ-MCKP instance. We use this transformation because Engler et al. [2] have already designed a pseudo-polynomial time algorithm for ϵ-MCKP which we want to use to solve ϵ-EMCKP.

A key requirement to this transformation process is that all atoms of an equivalence set must have equal weights and profits. Atoms grouped into equivalence sets by their neighborhoods automatically fulfill this requirement since their subgraphs have the same isomorphism class and the atoms then get the same weights and profits.

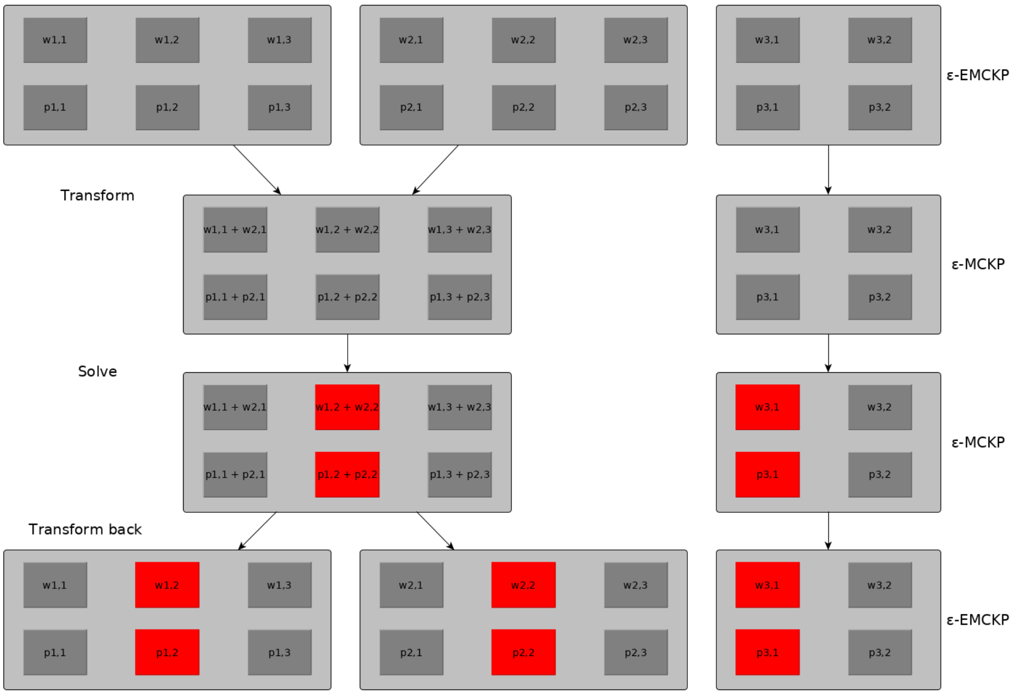

Figure 3: Schematic process of solving an ϵ-EMCKP instance by transforming it into an ϵ-MCKP instance, solving that and transforming the solution to the ϵ-EMCKP instance. The first two classes were chosen as equivalent. Marked in red is a selected solution

An ϵ-EMCKP instance I can be transformed into an ϵ-MCKP instance I' by combining all classes in an equivalence set to a single class by summing up their weights and profits element-wise. In any feasible solution to ϵ-EMCKP, the same item must be selected on all classes of an equivalence set and thus we can combine these items that always have to be selected together without compromising on feasibility and optimality of solutions.

Algorithm 1 Transform ϵ-EMCKP to ϵ-MCKP

```
for $E_l \in \{E_1, \dots, E_k\}$ do
    for $i \in E_l$ do
        for $j \in N_i$ do
            $w'_{l,j} \leftarrow w'_{l,j} + w_{i,j}$
            $p'_{l,j} \leftarrow p'_{l,j} + p_{i,j}$
    create class $N'_l$ with items $j = (w'_{l,j}, p'_{l,j})$
$\epsilon' \leftarrow \epsilon$
$c' \leftarrow c$
return $I' = (\{N_1, \dots, N_k\}, c', \epsilon')$
```

The resulting ϵ-MCKP instance I' now only has one class N'_l for every equivalence set

E_l of the ϵ-EMCKP instance. Capacity c and error ϵ stay the same between I and I'.

I' can then be solved with any algorithm suitable for ϵ-MCKP. We call that solution S' in the following.
To build a solution S for the ϵ-EMCKP instance I, we do the opposite process of the transformation: If item j has been selected on class N'_l in S' we need to select item j in all classes N_i where $i \in E_l$ (select item j in all classes that are combined into N'_l). This process does not alter the score (sum of profits) or total charge (sum of charges) of a solution, as any selected item j in S' consists of the items j that gets selected in S.

Theorem 5.1. Let I be an ϵ-EMCKP instance and I' be the ϵ-MCKP instance that is the result of applying Algorithm 1 on I. Let S' be a solution to I' and S be the solution to I that is build from S' by reversing Algorithm 1. Then I and I' are equivalent:

$$S' \text{ is a feasible solution to } I' \Leftrightarrow S \text{ is a feasible solution to } I$$

$$S' \text{ is an optimal solution to } I' \Leftrightarrow S \text{ is an optimal solution to } I$$

The proof can be found in Appendix A.

We introduced an algorithm to transform ϵ-EMCKP instances into ϵ-MCKP and have shown that ϵ-EMCKP can be solved optimally with any algorithm that solves ϵ-MCKP optimally.

5.2.1 Improving the transformation algorithm

Summing up all weights element wise as implemented in Algorithm 1 requires looking at every single weight in the ϵ-EMCKP instance. This has a worst case running time of $\mathcal{O}(m \cdot |N_{max}|)$ where m is the number of classes and $|N_{max}|$ is the size of the largest class, since each class is contained in exactly one equivalence set.

This algorithm and the running time can be further improved upon by incorporating the fact that all classes of an equivalence set have to have element-wise identical weights and profits. This allows for not looking at each weight and profit of every class, but only looking at each weight and profit of the first class of every equivalence set.

Instead of element-wise addition of the same weight/profit over and over again, we can simply multiply the weight/profit of the first (or any) class that is contained in the current equivalence set by the times it gets added up which is the number of classes contained in that current equivalence set:

$$w'_{l,j} = \sum_{i \in E_l} w_{i,j} = |E_l| \cdot w_{l_1,j}$$

Algorithm 2 Improved Transformation of ϵ-EMCKP to ϵ-MCKP

```
for $E_l \in \{E_1, \dots, E_k\}$ do
    $i \leftarrow E_{l_1}$
    for $j \in N_i$ do
        $w'_{l,j} \leftarrow w_{i,j} * |E_l|$
        $p'_{l,j} \leftarrow p_{i,j} * |E_l|$
    create class $N'_l$ with items $j = (w'_{l,j}, p'_{l,j})$
$\epsilon' \leftarrow \epsilon$
$c' \leftarrow c$
return $I' = (\{N_1, \dots, N_k\}, c', \epsilon')$
```

Given the same input instance, Algorithm 1 and Algorithm 2 produce the same output instance.

Algorithm 2 has a worst case running time of $\mathcal{O}(k \cdot |N_{max}|)$ where k is the number of equivalence sets and $|N_{max}|$ is the size of the largest class. This improved algorithm also uses fewer read accesses.

If we have fewer equivalence sets than classes ($k < m$) then Algorithm 2 is faster than Algorithm 1.

If we do not have fewer equivalence sets than classes then each equivalence set has exactly one element and k equals m. Then we can omit the equivalence sets completely, skip the transformation and solve ϵ-MCKP because in every class the selected charge only has to be identical to itself (like in Theorem 4.2).

Algorithm 2 is always favorable to Algorithm 1, because it is always faster unless $k = m$, in which case we can skip the transformation altogether.

5.3 LP Relaxation

As we have shown, ϵ-EMCKP is a weakly NP-complete (or weakly NP-hard in the optimization variant) problem.

Solving the linear program relaxation is a common way of simplifying an NP-hard integer linear program by dropping the integrality constraint and allowing the variables to be continuous.

In this case, to relax the ϵ-EMCKP ILP, the 0/1 constraint

$$x_{i,j} \in \{0, 1\} \text{ for } 1 \leq i \leq m,\ j \in N_i$$

is replaced by allowing the variables to be float values in the range $[0, 1]$:

$$0 \leq x_{i,j} \leq 1 \text{ for } 1 \leq i \leq m,\ j \in N_i$$

To get a charge for each atom i of a molecule, instead of choosing the single charge $w_{i,j}$ where $x_{i,j} = 1$, the sum of charges $\sum_{j \in N_i} x_{i,j} w_{i,j}$ yields the charge to atom i with the sum of profits $\sum_{j \in N_i} x_{i,j} p_{i,j}$ being its score.

By relaxing the integrality constraint we are able to freely combine items within a class. The symmetric charge assignment problem and charge assignment problem by Engler et al. [2] can be solved efficiently, if we allow the partial charges to deviate from the reference charges.

5.4 Mean, Median or Mode Selection

There are three more alternative baseline methods described by Engler et al. [2] for assigning partial charges to the atoms of a query molecule. For every atom, the partial charge is calculated as an average of the collected charges of that neighborhood. The charge for each atom gets assigned independently from the other atoms.

- Mean: The arithmetic mean of the collected charges is the assigned charge, calculated by dividing the sum of charges by the number of charges.
- Median: The middle value of all collected charges (ordered from lowest to highest) is the assigned charge. If there is an even number of charges, the mean of the two middle values is the median charge.
- Mode: The mode of the histogram (the charge that appears most often) is the assigned charge. On multimodal histograms, the mode closest to the median is the selected mode.

With these baseline methods of selecting a partial charge as some type of average, assignment of symmetric charges happens automatically. All atoms with identical neighborhoods get identical charge distributions and for each of those atoms, the same value gets selected (as the charge distributions all share the same mean/median/mode value).

These methods are therefore capable of assigning identical charges to symmetric atoms, but they do not necessarily deliver a solution to the symmetric charge assignment problem. Like described by Engler et al. [2], the sum of partial charges deviates from the required total charge, because individual charge assignment errors get accumulated.

6 Implementation

To solve ϵ-EMCKP we have two options: We can either solve the ILP or we can first reduce the classes with the transformation to ϵ-MCKP and then solve that ϵ-MCKP instance with any suitable method. Both options are implemented in Python 3.7.3 as additional classes in the implementation by Engler and Veen [6], called charge_assign. The source code is located in the charge_assign GitHub repository [7]. The LP-relaxation has also been implemented as a modification of the ϵ-EMCKP integer linear program.

ϵ-EMCKP can be used in the same way as the original ϵ-MCKP implementations and atoms automatically get grouped into equivalence sets after the collections of charges, requiring no additional manual steps. As a way of solving the ϵ-MCKP instance, the already implemented dynamic programming approach gets used. Since that can be solved with the programming languages C and Python, both are available options for solving ϵ-EMCKP as well.

6.1 Requirements

To run the tool, a python environment with the following packages is required:

- networkx
- msgpack-python
- numpy
- nauty

A key component to the implementation is nauty [5] as it is responsible for computing the canonical keys and matching the neighborhood induced subgraphs to the repository of molecules.

6.2 Usage

After installation or update of the charge_assign library, we can use ϵ-EMCKP by choosing the 'symmetric' version of the available chargers and assigning charges to molecules with it. Chargers are different Python classes that employ the different assignment methods.

7 Results

As a source of reference charges we used a snapshot of the ATB that contained 261.336 molecules. Only molecules with less than 50 atoms were included, as molecules with atom counts greater than 50 have not had higher level quantum mechanics calculations performed on them and thus were deemed to be too inaccurate to serve as reference charges.

From these 261.336 molecules there were 151.444 molecules that were fully covered by a fixed $k = 3$ neighborhood and for which ϵ-EMCKP was solvable. Fully coverable means that for every atom of the query molecule at least one molecule with an isomorphic k-neighborhood induced subgraph was found in the repository.

There were a total of 6 molecules, for which ϵ-MCKP was solvable but ϵ-EMCKP was not solvable and another 7 molecules for which partial charges could not be determined with either ϵ-MCKP or ϵ-EMCKP but for which partial charges could successfully be assigned using the simpler methods of selecting the mean, median or mode.

7.1 Methodology

To evaluate our methods, we have used the same leave-one-out-analysis that has been used by Engler et al. [2] and supplied with the implementation by Engler and Veen [6]. For each query molecule, a repository that was filtered of all molecules isomorphic to the query molecule, was created to simulate the assignment of charges for new molecules not included with the database. With these filtered repositories the charges for all atoms of the fully coverable molecules were determined. Afterwards, the distance between the computed charge and the ATB assigned charge was calculated for each atom to determine the charge error.

We opted for a fixed $k = 3$ shell size as the neighborhood size because Engler et al. [2] have used a fixed $k = 3$ neighborhood as well. Martin Engler (personal communication, June 18, 2019) explained that selecting the right shell size is a trade-off between coverage and accuracy as a lower shell size provides a higher probability to cover the whole query molecule while a higher shell size results in more accurate charges. He said that a shell size of 3 is a sweet spot, as it allows fully matching aromatic rings which gives a large boost in accuracy compared to shell sizes of 2, 1 or 0.

Fully matching an aromatic ring means that on Benzene-like rings (and all rings of up to 6 atoms) the 3-neighborhood induced subgraph of every atom in a ring includes the whole aromatic ring and attached side chains (apart from a potential side-chain on the opposite sided atom). Charges collected from reference molecules have to belong to a very similar ring structure and should in return be quite accurate. Matching the whole ring is important as ring structures within molecules are special structures that behave differently from other structures.

The maximal error ϵ was set to 0.01 to allow some freedom in choosing the charges, whilst keeping the maximal distance from the total charge low. We decided on this error-value, as it was the default setting in the implementation by Engler and Veen [6] and we had no reason to change it so something bigger or smaller in most cases. We only changed $\epsilon = 0.01$ to $\epsilon = 0$ in the analysis of the LP-relaxation method, since charges can be combined there and we expected that hitting an exact total charge would not compromise on the charge accuracy as much as in ϵ-EMCKP.

7.2 ϵ-EMCKP

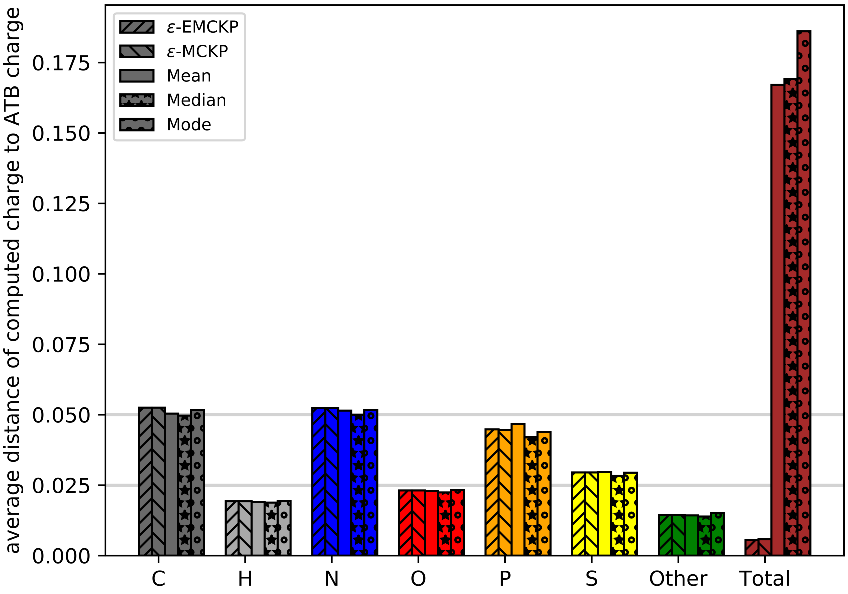

Figure 4: Results of the leave-one-out analysis with a fixed $k = 3$ neighborhood, showing the average distance between the computed charges to the charge found in the ATB. ϵ has been set to 0.01. GROMOS IACM types have been used for atom type labeling. Average distances are grouped by atom type. Column 'Total' is the total charge of the molecule (sum of all partial charges).

Average charge distances of ϵ-EMCKP and ϵ-MCKP are nearly identical on all atom types. Molecules charged with ϵ-EMCKP and ϵ-MCKP only differ on a handful of atoms, sometimes the assigned charges are completely identical, as ϵ-MCKP will assign identical charges if it happens to be the highest scoring solution. In the other cases where the methods produced different charges on equivalent atoms, charges on most atoms are still identical between ϵ-EMCKP and ϵ-MCKP and just a few charges are adjusted between ϵ-MCKP and ϵ-EMCKP.

Apart from the total charge difference of the molecule, phosphorus shows the largest gaps between the different charge-methods. This difference between methods is explained by the number of phosphorus atoms for which charges were assigned, as there were only 3207 phosphorus atoms on the 151.444 evaluated molecules. In contrast, there were 1.927.400 partial carbon atom charges assigned. The average charge error on phosphorus is low nonetheless and we attribute this to the few reference charges that were found for phosphorus being very specific and accurate to the chemical environment whilst there is a large range of different charges for carbon atom neighborhoods.

As with the results of Engler et al. [2], the method of selecting the median charge still produces the charges with the lowest average distance to the ATB assigned charge, but with the total charge showing a large distance to the actual total charge.

We will note here that, as with all these charges from the ATB, there is no definitive right or wrong partial charge. Force fields and different models all have their own assumptions to physics and assign different partial charges. For Benzene, there are currently two more possibilities of assigned charges in the ATB and probably many more in different force fields. But all possibilities of assigning realistic charges to Benzene will have uniform charges across carbon atoms and across hydrogen atoms in common.

7.3 LP-Relaxation

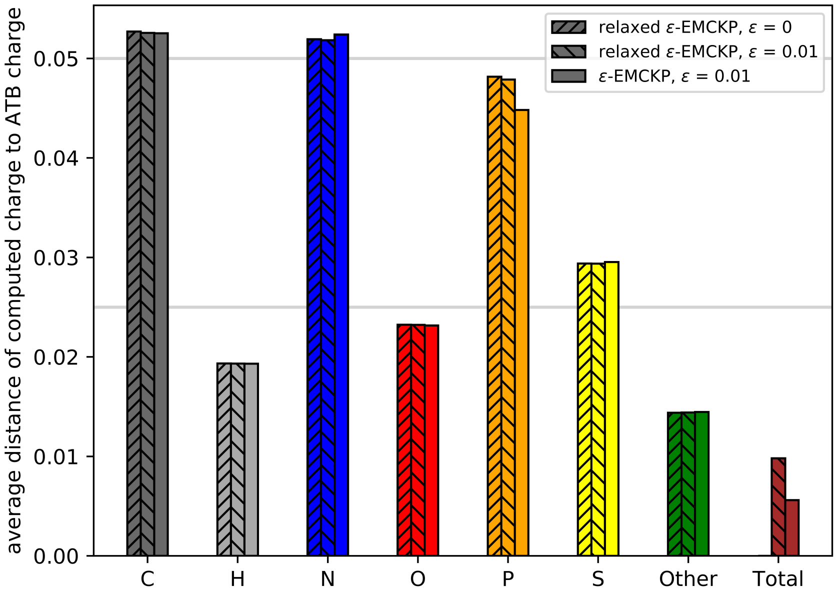

Figure 5: Results of the leave-one-out analysis with k = 3 showing the average distance between the computed charges to the charge found in the ATB. GROMOS IACM types have been used for atom type labeling. The relaxed ϵ-EMCKP has been run with $\epsilon = 0$ and with $\epsilon = 0.01$.

The average charge errors of the relaxation with $\epsilon = 0.01$ are close to the integer ϵ-EMCKP charge errors, with some atom types (C, H, O) having slightly worse averages and some (S, N, Other) being slightly better. These artificial combined charges are on average then not much better and not much worse than selecting straight from the QM-derived charges with ϵ-EMCKP. Thus the relaxation seem to provide reasonable estimates at the partial charges, like the other methods do as well, whilst reducing the theoretical complexity.

The relaxation of ϵ-EMCKP always produces scores that are better than those of the integer ϵ-EMCKP. An optimal solution of the integer ϵ-EMCKP is always a solution to the relaxed ϵ-EMCKP, but an optimal relaxed ϵ-EMCKP solution might not be an integer ϵ-EMCKP solution. This higher score comes at the cost of an increased deviation from the total charge if allowed. Notice that almost every molecule has a total charge difference of 0.01 in the relaxation with $\epsilon = 0.01$, as the average total charge difference is about 0.0098. This means that the total charge of a molecule charged with the relaxation method will be close to the edge of the allowed difference almost every time.

Restricting charges to match the exact total charge ($\epsilon = 0$) mostly affects carbon, nitrogen and phosphorus atoms. This can be once again explained by a low number of phosphorus atoms and a large range of carbon charges. It is important to note that even though the nitrogen average charge error is increased by decreasing ϵ to zero, it is still better than the average charge error of nitrogen in ϵ-EMCKP and ϵ-MCKP.

7.4 Atom label alphabet

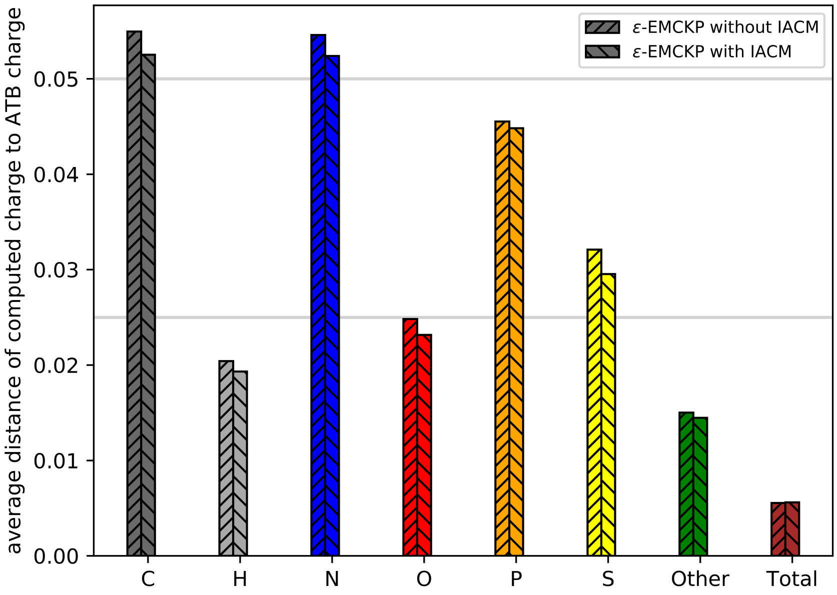

Figure 6: Results of the leave-one-out analysis with k = 3 showing the average distance between the computed charges of ϵ-EMCKP to the charge found in the ATB. Atom labels consist of normal atom types (periodic table naming) and GROMOS IACM atom labels.

GROMOS IACM atom labels provide a boost to the accuracy of the assigned charges as the isomorphism classes get more specific to the local atom neighborhood. We suspect that the reason for this accuracy improvement is that the IACM labels are assigned according to the neighbors and bond types between the atoms. In some cases these IACM labels might include information about atoms that are not part of the k-neighborhood but have an influence on the partial charge.

Using IACM types as atom labels sometimes comes at the cost of not being able to cover a molecule, as 167.710 molecules were fully covered with periodic table atom labels and

151.444 molecules were fully covered with IACM atom labels. We assume that on the molecules that were not fully covered with a fixed $k = 3$ neighborhood size and IACM labels, the assigned IACM labels were too specific to find a match for in the repository of molecules.

Since they provide a boost in accuracy, the IACM atom labels should be used where possible.

7.5 Limitations of the atom-neighborhood driven approach

The molecules decane (molID 2589) and to even more extreme extent hexadecane (molID 6329) are examples for where the neighborhood driven approach (in ϵ-MCKP and ϵ-EMCKP) reaches its limits and fails in accurate equivalence set detection.

Both molecules are simple alkanes: single bonded carbon chains surrounded by hydrogen atoms (called hydrocarbon chains). Alkanes are totally symmetrical molecules that are mirrored at the center of the molecule and can consist of arbitrarily many repeated CH_2 sections between the two CH_3 ends.

If a shell size of $k = 3$ is chosen as the neighborhood size then for example on decane, the four innermost carbon atoms (C4 - C7) are put into the same equivalence set. According to the ATB this equivalence set should be split into two equivalence sets. See Fig. 7.

The k-neighborhood induced subgraphs of the four innermost carbon atoms are all isomorphic on a $k = 3$ neighborhood size, since that size is not large enough to include the three hydrogen atoms connected to the C1 or C10 atom. This structure (three hydrogen atoms connected to one carbon atom) only occurs at the end of an alkane. It functions as somewhat of a reference point to an atoms location within the molecule. Because this structure is not included in any of the C4 to C7 induced neighborhood graphs, the position of those four innermost carbon atoms, relative to an end of the molecule, can not be fixed and in return they are grouped into the same equivalence set.

A similar observation, but on a larger scale, is made when assigning charges to hexadecane (or much larger alkanes) with a $k = 3$ neighborhood size.

As noted by Engler et al. [2], there is a trade off between choosing a larger neighborhood size k for more specific charges that comes at cost of the number of different charges to choose from. The larger alkanes we look at, the higher we have to increase the neighborhood size to get the actual equivalence sets. Since we can think of arbitrarily large alkanes, apart from not having enough computational power to compute large equivalence sets, we can reach a point where there are no large enough reference molecules in the database anymore. Then equivalence sets and resulting charges are reduced to a lower size and a similar pattern to the shown pattern on decane will emerge.

This example on alkanes illustrates that equivalence detection on with a neighborhood-based approach can lead to a false detection of equivalent atoms if the neighborhood size k is too low. In combination with ϵ-EMCKP this can result in false enforcement of identical charges.

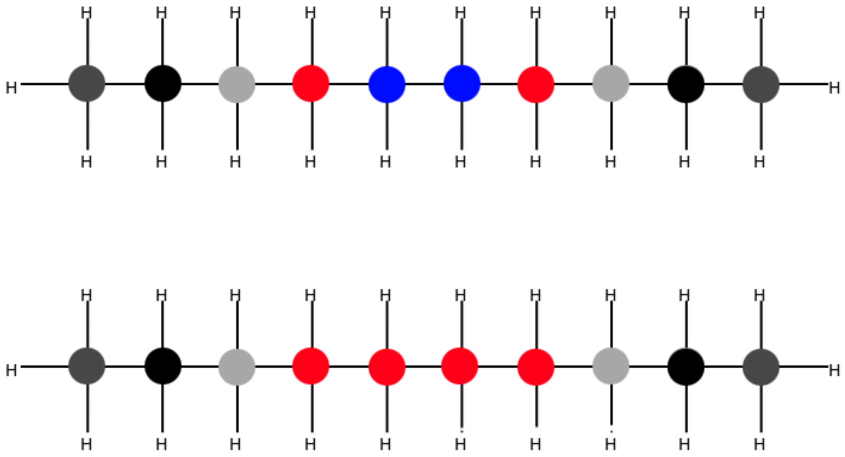

Figure 7: Visual equivalence set representation of carbon atoms in decane (molID 2589). On the top are equivalence sets in the ATB and on the bottom are the equivalence sets found by the ϵ-EMCKP implementation. Equivalence sets of hydrogen atoms have been left out in this figure. All carbon atoms of one color are included in the same equivalence set. Marked in gray shades are the equivalence sets that are the same between the ATB and ϵ-EMCKP. Marked in blue is the additional equivalence set that exists in the ATB but is not found by ϵ-EMCKP with $k = 3$.

8 Discussion

Charges found by the ϵ-MCKP implementation and the ϵ-EMCKP implementation are often indistinguishable and vary only on a few atoms of a molecule, if they vary at all. As ϵ-MCKP most of the time assigns charges close to the largest mode of the charge distributions [2], ϵ-EMCKP also assigns charges that are close to the largest mode. Atoms with equivalent neighborhoods receive the same charge distributions and often then the same charge is selected both by ϵ-MCKP and ϵ-EMCKP. On the random sample molecules we used during testing, most atoms of the molecules were assigned the same charge both by ϵ-MCKP and ϵ-EMCKP and only a fraction of the atoms showed different values at all. This also explains the similar average charge errors between ϵ-MCKP and ϵ-EMCKP.

We conclude from this that the charge distributions often already indicate which atoms have identical neighborhoods. The effect of this can be seen when solving ϵ-MCKP without restrictions to the total charge c and error margin ϵ with any of the baseline methods. Since each atom is independent of other atoms to select a charge, all atoms with identical neighborhoods and therefore identical charge distributions already get the same charge. Assigning identical charges is natural to the process of reference charge collection from isomorphic neighborhoods, it is just the ϵ-MCKP constraint of the partial charge sum to the total charge that sometimes results in different charges on symmetric atoms. ϵ-EMCKP successfully combines the symmetric charges from the baseline-methods with the partial charge sum constraint of ϵ-MCKP.

Charges assigned with ϵ-EMCKP are also more reliable and consistent with changing data, as symmetric atoms will always receive identical charges, whereas with ϵ-MCKP, charges on chemically equivalent atoms might be identical on one set of data and not identical on the next set of data. As new molecules are added every day to the ATB, this presents another advantage of ϵ-EMCKP over ϵ-MCKP.

Figure 8 shows the charges assigned to Benzene with ϵ-EMCKP. Not only are the partial charges assigned to the carbon atoms now with ϵ-EMCKP uniform across all atoms, but the distance of every charge to the ATB-assigned charge is smaller than the distance of the ϵ-MCKP assigned charge to the ATB, even though both methods use the same repository of data. This shows that the charges assigned with ϵ-EMCKP are more reliable than the ϵ-MCKP assigned charges.

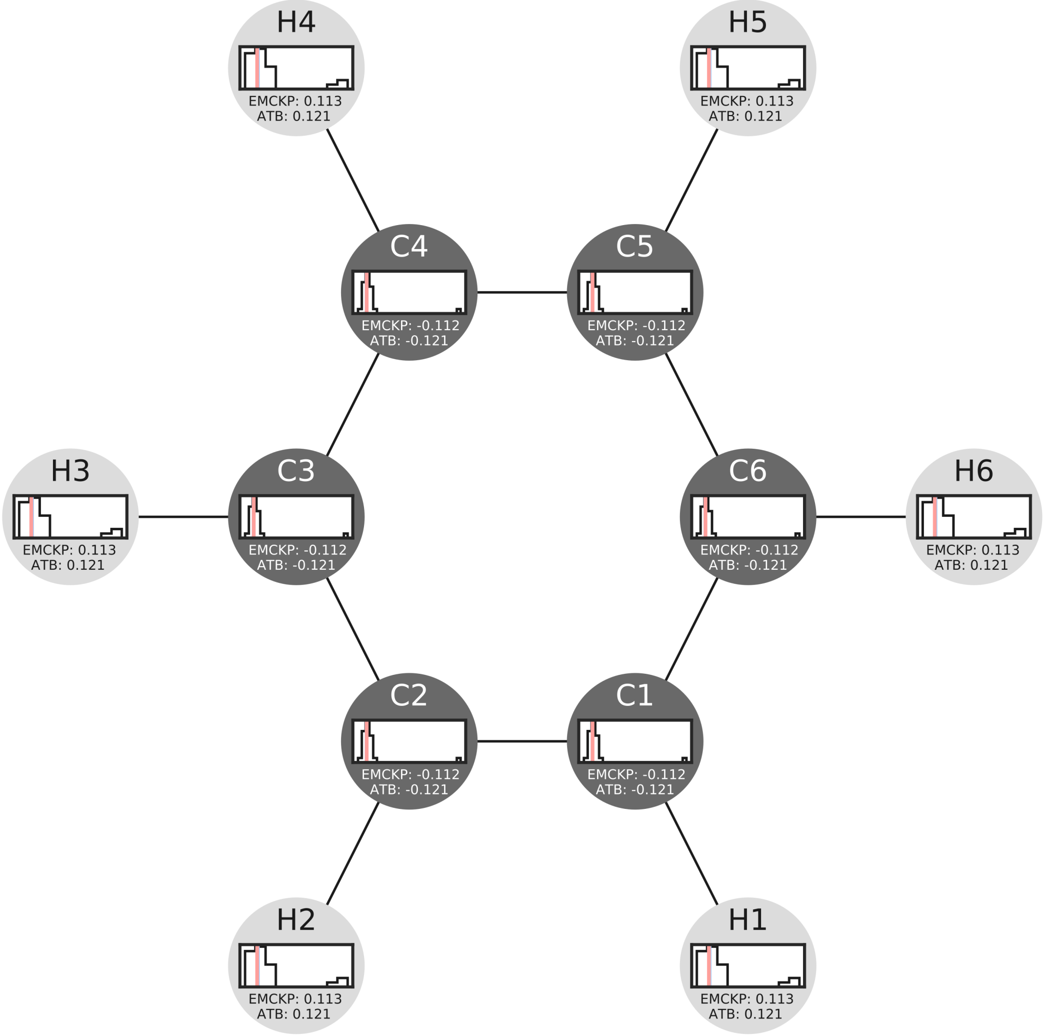

Figure 8: Benzene (molID 342920) charges assigned with ϵ-EMCKP

Table 1: Average distance of the computed partial charges to the ATB assigned charge on Benzene (molID 342920).

Atom type	ϵ-MCKP	ϵ-EMCKP
C	0.0173	0.009
H	0.016	0.008

9 Outlook

What has yet to be analyzed is running time and scaling of ϵ-EMCKP and accuracy on large molecules (> 50 atoms). Since the ϵ-MCKP DP scales linearly with the number of items and the blowup factor and transformed capacity [2], with our transformation of ϵ-EMCKP to ϵ-MCKP we cut down on running time since we reduce the number of items, while the blow-up factor and transformed capacity stay the same. We noticed a decrease in running time of the dynamic programming during manual testing. We expect an overall decrease in running time on most molecules, as most molecules have at least some chemically equivalent hydrogen atoms.

Engler et al. [2] have noticed partial charges on the outer atoms to be assigned well with more buried atoms not receiving as accurate charges. We expect this trend to continue with ϵ-EMCKP, as the deeper an atom is buried, the less likely we think it will be to find an identical neighborhood within the same molecule, while on the other hand the outer-most atoms often are hydrogen atoms for which identical neighborhoods often are found in the same molecule.

10 Summary

We have shown that the ϵ-MCKP approach does not always mirror the symmetry of neighborhoods within a molecule, as it lacks the necessity to do so. We have build a modification of ϵ-MCKP to always assign symmetry-mirroring charges and showed two algorithmic ways of solving it, one with Integer Linear Programming and one with a transformation to ϵ-MCKP. We also described LP-Relaxation and the average-based approaches as alternatives to solving the symmetric charge assignment problem.

We added ϵ-EMCKP to the ϵ-MCKP implementation in Python and we then compared the results in a leave-one-out evaluation to the existing ways of solving the charge assignment problem by using a snapshot of the ATB database as reference and validation data. We have shown that, on average, charges assigned with ϵ-EMCKP have a similar average distance to QM-derived charges as ϵ-MCKP assigned charges but charges assigned with ϵ-EMCKP are better suited for MD simulations that involve molecules with symmetric neighborhoods. When considering a molecule in its entirety, ϵ-EMCKP assigned charges are more reliable than ϵ-MCKP assigned charges, as the charges are always uniform across equivalent neighborhoods. This is especially noticeable on molecules that are highly symmetrical or consist of multiple identical subsections, as ϵ-MCKP sometimes assigns strange charges that do not fit to the other assigned charges.

11 Acknowledgments

I am very grateful to Gunnar Klau for making this thesis possible and always taking time. I am also grateful to all colleagues for the valuable discussions, and to Martin Engler and Lourens Veen for answering my questions about the paper and implementation, providing the additional code to create Figures 1 and 8, and for proofreading. I thank Martin Stroet for providing access to the ATB and sending the molecular data.

References

[1] Alpeshkumar K. Malde et al. "An Automated Force Field Topology Builder (ATB) and Repository: Version 1.0". In: *Journal of Chemical Theory and Computation* 7.12 (2011). PMID: 26598349, pp. 4026–4037. DOI: 10.1021/ct200196m. eprint: https://doi.org/10.1021/ct200196m.

[2] Martin S. Engler et al. "Automated partial atomic charge assignment for drug-like molecules: a fast knapsack approach". In: *Algorithms for Molecular Biology* 14.1 (Feb. 2019). ISSN: 1748-7188. DOI: 10.1186/s13015-019-0138-7.

[3] Hans Kellerer, Ulrich Pferschy, and David Pisinger. "Knapsack Problems". In: Berlin Heidelberg: Springer-Verlag, 2004. ISBN: 978-3-642-07311-3. DOI: 10.1007/978-3-540-24777-7.

[4] Martin S. Engler et al. "Enumerating common molecular substructures". In: *PeerJ Preprints 5:e3250v1* (2017). DOI: 10.7287/peerj.preprints.3250v1.

[5] Brendan D. McKay and Adolfo Piperno. "Practical graph isomorphism, {II}". In: *Journal of Symbolic Computation* 60.0 (2014), pp. 94–112. ISSN: 0747-7171. DOI: 10.1016/j.jsc.2013.09.003. URL: http://www.sciencedirect.com/science/article/pii/S0747717113001193.

[6] Martin S. Engler and Lourens Veen. *charge assign*. Oct. 2018. DOI: 10.5281/zenodo.1475888. URL: https://doi.org/10.5281/zenodo.1475888.

[7] *charge assign Github Repository*. Visited: 2019-07-22. URL: https://github.com/MD-Studio/charge_assign.

12 Appendix A

Table 2: Average distance of computed charge to ATB charge, rounded to 6 decimal places. Data shown in figure 4

Type	ϵ-EMCKP	ϵ-MCKP	Mean	Median	Mode
C	0.052527	0.05253	0.050396	0.049726	0.051636
H	0.019319	0.019327	0.01911	0.018848	0.019391
N	0.052398	0.052333	0.05148	0.050005	0.051743
O	0.023166	0.023165	0.022918	0.022438	0.02327
P	0.044815	0.044534	0.046735	0.042156	0.043853
S	0.02954	0.02955	0.029771	0.02833	0.029446
Other	0.014467	0.014453	0.014274	0.013915	0.015179
Total	0.005609	0.005819	0.167109	0.169188	0.186075

Table 3: Average distance of computed charge to ATB charge, rounded to 6 decimal places. Data shown in figure 5

Type	relaxed ϵ-EMCKP with $\epsilon = 0$	relaxed ϵ-EMCKP with $\epsilon = 0.01$	ϵ-EMCKP
C	0.052697	0.052565	0.052527
H	0.019336	0.019332	0.019319
N	0.051924	0.051834	0.052398
O	0.023235	0.023221	0.023166
P	0.048164	0.04788	0.044815
S	0.029386	0.029368	0.02954
Other	0.014392	0.014414	0.014467
Total	0.0	0.009809	0.005609

Table 4: Average distance of computed charge to ATB charge, rounded to 6 decimal places. Data shown in figure 6

Type	ϵ-EMCKP without IACM	ϵ-EMCKP with IACM
C	0.054949	0.052527
H	0.020414	0.019319
N	0.054593	0.052398
O	0.024824	0.023166
P	0.045525	0.044815
S	0.032102	0.02954
Other	0.015016	0.014467
Total	0.005544	0.005609

Proof. Proof of theorem 5.1
Let I be an ϵ-EMCKP instance and I' be the ϵ-MCKP instance that is the result of applying Algorithm 1 on I. Let S' be a solution to I' and S be the solution to I that is build from S' by reversing Algorithm 1.

- S' is a solution to $I' \implies S$ is a solution to I

  A valid solution to ϵ-EMCKP requires the selected items of all atoms in an equivalence set to be the same. By first combining all atoms of an equivalence set together and later selecting the same item on all atoms of that equivalence set we satisfy that condition. Due to the sum of weights and sum of profits, the capacity and the error being unchanged in the transformation and S' having to be a valid solution, S has to be a valid solution for I.

- S' is an optimal solution to $I' \implies S$ is an optimal solution to I

  If S' is an optimal solution to I' and S is not an optimal solution to I then there must exist an optimal solution S_{opt} to I. Therefore there must exist a solution S'_{opt} to I' that can be transformed into S_{opt}. S'_{opt} must have a higher score than S', because the score of a solution does not change and S_{opt} has a higher score than S, whilst S has the same score as S'. This is a contradiction to S' being an optimal solution to I' and as such, S has to be an optimal solution to I if S' is an optimal solution to I'.

- S' is not a solution to $I' \implies S$ is not a solution to I

  If S' is not a valid solution, then either multiple items (or no items) were selected in one class or the sum of weights is not within the range $c - \epsilon, c + \epsilon$. In the first case, S will not be a solution to I, as then multiple items (or no items) would be selected as well in some classes. In the second case, S will not be a solution to I too, because capacity, ϵ and sum of selected weights are not changed between I and I'.

- S' is not an optimal solution to $I' \implies S$ is not an optimal solution to I

  If S' is not an optimal solution to I', then there must exist an optimal solution S'_{opt} to I'. As we have already shown, the solution S_{opt} we get by transforming S'_{opt} is an optimal solution to I. S is not an optimal solution to I, as the score of S_{opt} is higher, because the score of S'_{opt} is higher than the score of S' and the score of a solution does not change with the transformation.

□