J Mol Model (2003) 9:235–241
DOI 10.1007/s00894-003-0126-0

ORIGINAL PAPER

S. F. Badreddin Abolmaali · Jörg K. Wegner · Andreas Zell

The compressed feature matrix—a fast method for feature based substructure search

Received: 29 October 2002 / Accepted: 3 February 2003 / Published online: 26 April 2003

Abstract The compressed feature matrix (CFM) is a feature based molecular descriptor for the fast processing of pharmacochemical applications such as adaptive similarity search, pharmacophore development and substructure search. Depending on the particular purpose, the descriptor may be generated upon either topological or Euclidean molecular data. To assure a variable utilizability, the assignment of the structural patterns to feature types is arbitrarily determined by the user. This step is based on a graph algorithm for substructure search, which resembles the common substructure descriptors. While these merely allow a screening for the predefined patterns, the CFM permits a real substructure/subgraph search, presuming that all desired elements of the query substructure are described by the selected feature set. In this work, the CFM based substructure search is evaluated with regard to both the different outputs resulting from varying feature sets and the search speed. As a benchmark we use the programmable atom typer (PATTY) graph algorithm. When comparing the two methods, the CFM based matrix algorithm is up to several hundred times faster than PATTY and when using the CFM as a basis for substructure screening, the search speed is accelerated by three orders of magnitude. Thus, the CFM based substructure search complies with the requirements for interactive usage, even for the evaluation of several hundred thousand compounds. The concept of the CFM is implemented in the software COFEA.

Keywords Substructure search · Descriptor · Features · Computer chemistry · Screening

Abbreviations *CFM:* compressed feature matrix · *MCS:* maximum common substructure · *HSCS:* highest scoring common substructure · *SSSR:* smallest set of smallest rings · *ESER:* essential set of essential rings · *ESSR:* extended set of smallest rings · *GSCE:* graph of smallest cycles at edges · *PATTY:* programmable atom typer · *HTS:* high throughput screening

S. F. B. Abolmaali (✉) · J. K. Wegner · A. Zell
Department of Computer Science,
University of Tuebingen,
Sand 1, 72076 Tübingen, Germany
e-mail: abolmaali@informatik.uni-tuebingen.de
Tel.: +49 7071 29 78979

Introduction

Substructure searching is a widely used method in pharmaceutical research. Its main fields of application are (a) the calculation of substructure descriptors such as structural keys, [1] hashed fingerprints, [2] molecular holograms [3, 4] and atom pairs, [5] (b) structure modification, e.g. in the usage of protonation models, and (c) maximum common substructure (MCS) analysis which is also commonly applied to problems of similarity evaluation. [6] Although all of these applications are based on the method of substructure searching, the corresponding descriptors are calculated from different prerequisites. As a rule, substructure descriptors and structure modification descriptors are generated on the basis of predefined structural patterns. Searching for the corresponding molecular substructures is performed using graph algorithms like the Ullmann subgraph isomorphism algorithm. [7] Therein, the respective patterns are commonly specified by SMARTS [8] strings. If unsaturated compounds are evaluated, a distinction between subgraph and substructure searches becomes necessary. [9] For a real substructure search, the defined patterns must include information about both the comprised chemical elements and the respective binding types. In contrast, neglecting the exact determination of bonds results in the recognition of matching subgraphs. For efficient usage of a graph based substructure search, a preceding ring detection algorithm is required, such as smallest set of smallest rings (SSSR), [10] essential set of essential rings (ESER), [11] extended set of smallest rings (ESSR) [12] and graph of smallest cycles at edges (GSCE). [13] Further acceleration may be achieved by sorting the search patterns in such a way that rare atoms are checked first during pattern assignment. In contrast to substructure and structure modification descriptors, calculating the MCS of two molecules is independent of redefined patterns. Instead, the largest possible structure occurring in both compounds is recognized only with respect to the sets of substructures/subgraphs that are actually found in the two compounds. Again, the distinction between matching substructures or subgraphs is due to the specification of binding types.

In this work, we introduce the compressed feature matrix (CFM) as a molecular descriptor for subgraph/substructure search. Particularly in pharmaceutical research, a main goal of substructure search is to detect molecular compounds showing similar biological effects as known active ingredients. Therein, understanding the contribution of a certain substructure to the particular effect, the features of atoms and atomic groups (e.g. charge, hydrophobicity, aromaticity, etc.) are commonly more significant than the respective chemical elements. Accordingly, the CFM descriptor was designed to focus on the feature graphs of the evaluated compounds. To make the method adaptive to varying questions, the respective feature sets may be arbitrarily composed by the user. And since the CFM correlates the determined features by their topological or Euclidean distances, it represents the complete feature graph information of a given structure. Therefore, the problem of subgraph/substructure search can be solved by matching the CFM of a query substructure to the one of the tested molecule. In this respect, the main condition that a defined feature set has to fulfill is that it is suitable to completely describe the desired elements of the searched substructure.

Besides the method of substructure searching, the CFM is used in further pharmacochemical applications as pharmacophore development and similarity searches. [14] In the following sections, the structure of the CFM as well as the CFM based method of subgraph/substructure search are described. The method is evaluated with regard to both the different results achieved by three particular feature sets and the search speed. Furthermore, the CFM search is compared to the programmable atom typer (PATTY) backtracking algorithm, [15] which is a graph algorithm for substructure search and included in the software package JOELib. [16, 17] The same algorithm was used for the generation of the CFMs.

Materials and methods

CFM structure

The core of the CFM is either a distance or a geometry matrix, depending on whether it is constructed upon topological or geometrical molecular data. As an advancement of the common matrices, the CFM is not restricted to the representation of atoms but may be built on the basis of any (user defined) set of structural features. Since changed feature sets cause altered matrices, the features that occur in a molecule are an integral part of its CFM, which is therefore defined as

$$\mathbf{C}: \begin{pmatrix} \mathbf{f} \\ \mathbf{D} \end{pmatrix} \tag{1}$$

where the row vector $\mathbf{f}:=(F_k)_{k=1}^{n}$ contains the features F, and $\mathbf{D}:=(d_{ij})_{i,j=1}^{n}$ is the respective distance or geometry matrix. In the following, the single features F are typed in upper case letters, while the kinds and groups of features f are in lower case.

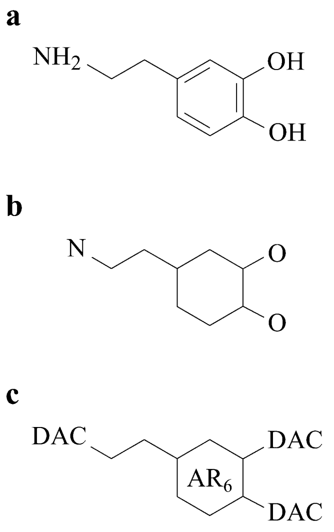

Fig. 1a–c Chemical structure (**a**) and feature graphs (**b**, **c**) of dopamine

a

	N	C	C	C	C	C	C	C	C	O	O
N	0	1	2	3	4	5	6	5	4	6	7
C	1	0	1	2	3	4	5	4	3	5	6
C	2	1	0	1	2	3	4	3	2	4	5
C	3	2	1	0	1	2	3	2	1	3	4
C	4	3	2	1	0	1	2	3	2	2	3
C	5	4	3	2	1	0	1	2	3	1	2
C	6	5	4	3	2	1	0	1	2	2	1
C	5	4	3	2	3	2	1	0	1	3	2
C	4	3	2	1	2	3	2	1	0	4	3
O	6	5	4	3	2	1	2	3	4	0	3
O	7	6	5	4	3	2	1	2	3	3	0

b

	DAC	DAC	DAC	AR6
DAC	0	6	7	3
DAC	6	0	3	1
DAC	7	3	0	1
AR6	3	1	1	0

Fig. 2a, b Topological CFMs of dopamine based on different feature sets. **a** Atomic representation. **b** Feature set "C" (Table 1)

Figure 1 shows the chemical structure (a) and two topological feature graphs (b, c) of dopamine that are based on different feature sets.

The first feature set describes a molecule on its atomic level, representing it in the same way as a common distance matrix. In contrast to this, the second graph (c) results from a feature set that distinguishes between 15 different kinds of features (Table 1, feature set "C"): terminal carbon atoms (cat), hydrogen bond donors (don) and acceptors (acc), atoms that may occur as either donors or acceptors (dac), positive (pos) and negative (neg) sites, aromatic rings (ar3, ar5, ar6) and nonaromatic rings (r3–r8). Within this set the features don, acc and dac are based on a classification proposed by Markus Böhm and Gerhard Klebe. [18] Figure 2 shows the CFMs of dopamine based on these two feature sets.

Table 1 Three different feature sets, either regarding chemical elements without (A) and with (B) information about binding types, or biological properties (C). The description of the features is based on SMARTS

Feature set A “Chemical element”		Feature set B “Chemical element and binding type”		Feature set C “Biological property”	
Descr.	Feature	Description	Feature	Description	Feature
C	c_al	[CQ1X4]	c_al_1	[CH3,CQ1H2,CQ1H1]	cat
c	c_ar	[CQ1X3]	c_al_2	[$([NH2]-c),NQ1H3,NQ2H2,NQ3H1, NQ2H1,$(Cl-*),$(Br-*),$(I-*)]	don
O	o_al	[CQ1X2]	c_al_3	[OQ1X1,OQ2X2,NQ3X3,NQ2X2,NQ1X1]	acc
o	o_ar	[CQ2X4]	c_al_4	[$([NH2]-C),$([OH]-*)]	dac
N	n_al	[CQ2X3]	c_al_5	[+,++,+++]	pos
n	n_ar	[CQ2X2]	c_al_6	[-,–,—]	neg
		[CQ3X4]	c_al_7	*1**1	r3
		[CQ3X3]	c_al_8	*1***1	r4
		[CQ4X4]	c_al_9	*1****1	r5
		c	c_ar	*1*****1	r6
		[OQ1X2]	o_al_1	*1******1	r7
		[OQ1X1]	o_al_2	*1*******1	r8
		[OQ2X2]	o_al_3	a1aa1	ar3
		o	o_ar	a1aaaa1	ar5
		[NQ1X3]	n_al_1	a1aaaaa1	ar6
		[NQ1X2]	n_al_2		
		[NQ1X1]	n_al_3		
		[NQ2X3]	n_al_4		
		[NQ2X2]	n_al_5		
		[NQ3X3]	n_al_6		
		n	n_ar		

Since a CFM holds the entire topological or geometrical information of a molecule, the descriptor is invariant with respect to any kind of atom numbering, rotation and center of gravity transformations. Accordingly, the sequence of the features within the row vector **f** may in principle be determined arbitrarily. However, to standardize the procedure of substructure search, the particular feature types are automatically grouped according to an implicit order, depending on the succession of type definition.

Substructure search

The process of substructure search is performed in two steps. First, those database molecules that potentially contain the particular substructure are selected from the original test data set. In the second step, the actual search algorithm is performed on the preselected data. As a precondition of both modules, the query substructure as well as the tested molecules are represented by their CFMs.

Preselection

During preselection, each database molecule is analyzed by means of the maximum length of the substructure and of its feature composition. The maximum length is defined by the distance between the two furthermost features occurring in the substructure. Accordingly, the CFM of the tested molecule is searched for a corresponding entry matching both the distance value and the two respective feature types. If this essential feature pair is found, the row vector of the tested CFM is checked for the occurrence of those features F that the substructure is composed of. That means that for all feature types f occurring in the substructure S, the number of features $|F|$ of type f_i must be less than or equal to the number of features of the respective type contained in the tested molecule T:

$$|F|^{s}_{f_i} \leq |F|^{\tau}_{f_i}; \quad 1 \leq i \leq n_f. \tag{2}$$

Here, n_f is the total number of feature types found in the substructure. Only if both conditions are fulfilled is the substructure possibly a part of the tested molecule, and the latter is selected for further evaluation.

CFM based search algorithm

After preselection, the actual search algorithm is performed. Since this is based on the CFMs of both the query substructure and the tested molecule, the problem of substructure search leads back to the question whether a given submatrix **S** occurs in a test matrix **T**. For this evaluation, only the upper triangular matrices are considered, which is valid since the CFM descriptor is symmetrical. As above, the term “corresponding entries” of **S** and **T** always refers to corresponding subgraphs within the two respective molecules, i.e., the considered features as well as the interjacent distances are identical in both CFMs.

Due to the logic of indexing, the coordinates of all entries of a given matrix are deducible from the j values of its first row (in the nomenclature used, the rows and columns of a matrix are specified by the indices i and j, respectively). Given an $n \times n$ matrix, recombining the respective values by pairs yields the total n^2 different tuples (ij values) that represent the searched coordinates. In analogy, this procedure also permits the localization of the entries of an embedded submatrix. As an example, Fig. 3 shows a 4×4 submatrix that is placed within an 8×8 test matrix. Again, the positions of the submatrix entries (bold) are determined by the (underlined) column-indices found in the first row.

The algorithm of CFM based substructure search may be divided into three steps: first, those types of features that do not occur in **S**, and with them the appendant rows and columns, are removed from the test matrix **T**, since they have no influence on the detection of the query substructure. After that, both matrices comprise the same kinds of features, and each row of a matrix represents the connections of one particular feature (called target-feature in the following) to all other features that are relevant in the given context. In a second step, the entries of the first row of **S** are searched within each particular row of **T** that shows the respective target-feature. Note that the sequence of entries within the two compared rows is arbitrary. If the search is successful, the test matrix is reordered, placing the matching row at its top. Subsequently the residual coordinates are determined as described

00	01	**02**	03	04	**05**	**06**	07
10	11	12	13	14	15	16	17
20	21	**22**	23	24	**25**	**26**	27
30	31	32	33	34	35	36	37
40	41	42	43	44	45	46	47
50	51	**52**	53	54	**55**	**56**	57
60	61	**62**	63	64	**65**	**66**	67
70	71	72	73	74	75	76	77

Fig. 3 Exemplary distribution of submatrix entries within a test matrix

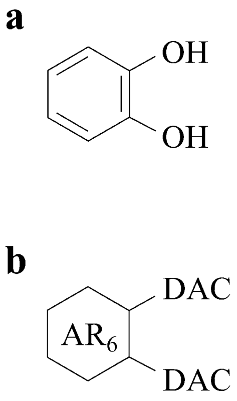

Fig. 4a, b Chemical structure (**a**), and feature graph (**b**) of benzene-1,2-diol based on feature set "C"

a

	DAC	DAC	AR6
DAC	0	3	1
DAC	3	0	1
AR6	1	1	0

b

	DAC	DAC	DAC	AR6
DAC	0	6	7	3
DAC	6	**0**	**3**	**1**
DAC	7	**3**	**0**	**1**
AR6	3	**1**	**1**	**0**

Fig. 5a, b Topological CFMs of benzene-1,2-diol (**a**) and dopamine (**b**). Within the latter, the comprised matrix of benzene-1,2-diol is highlighted

in the previous paragraph, and the entries of the resulting potential submatrix are compared to those of the query submatrix. A perfect match of these two matrices proves that **S** is indeed a submatrix of **T**. Accordingly, the tested molecule comprises the searched substructure.

As a basic example, Fig. 4 displays the chemical structure (a) and the feature graph (b) (based on feature set "C"), of benzene-1,2-diol. Figure 5 shows the respective CFM (a) and its appearance (bold) in the CFM of dopamine (b).

However, the algorithm described does not stop once the first appearance of the query submatrix is detected. Rather, the test matrix is successively searched for all occurrences of **S**. Thus, the result of the search algorithm is the number of substructure hits within the tested molecule.

Software

The concept of the CFM is implemented in the software COFEA which is written in Java 2, JDK version 1.4. Thus, being platform independent the program runs under various operating systems such as different Unix versions, Linux, and Microsoft Windows. Besides the described method of substructure search, the software provides two further pharmacochemical applications, pharmacophore development and similarity search (the latter was reported in a recent article [14]), which are also based on the concept of the CFM. The program package COFEA is used in the pharmaceutical research of the ALTANA Pharma AG and the Merck KGaA, where it is especially used for the search and optimization of lead structures, lead hopping and in high throughput screening (HTS) analysis.

Irrespective of the particular application, an essential step in using COFEA is the specification of the feature types that are desired to serve as a basis for the CFM representation of the analyzed molecules. Thereby, different structures showing similar biochemical properties are grouped within a single SMARTS string followed by the respective feature. As an example, the specification "[$([NH2]-C),$([OH]-C),$([OH]-c)] DAC" means that every structure that matches one of the three given SMARTS patterns (separated by commas) is represented by the feature "DAC" (hydrogen bond donor or acceptor) in the respective CFM. For the assignment of the features to the respective structural patterns the program COFEA provides an interface connection to the software package JOELib that uses the SSSR method for ring determination and the PATTY algorithm for the detection of patterns.

Following the specification of a feature set, the molecules that are to be analyzed are converted into their CFMs. As a basis for this, the respective structures must be available in a standard molecular format as e.g. the MDL Molfile format. [19, 20] The calculated CFM descriptors may optionally be stored in a file system or—due to an integrated database connection—in a database. On a Windows based computer with 512 MB RAM and a single 1,200 MHz AMD Athlon processor (which was used for all evaluations described below), the conversion of 100,000 structures took between 15 and 20 min, depending on the complexity of the determined features. Note that for a particular feature set, the CFMs of the given test molecules have to be calculated only once. All subsequent applications are directly performed on the stored CFM descriptors.

For a substructure search, two CFM files have to be specified, one of them containing the query substructure, the other one comprising the data set to be tested. Again, the respective compounds may alternatively be read from a database. As a search result, COFEA generates a sorted list (descending, most hits on top) of all tested molecules, which shows both the total number of substructure hits as well as the number of unique hits for each tested compound. As an example, searching for the substructure "benzene" within the structure of dopamine results in a total number of six hits, but only a single unique hit. This is because dopamine contains one benzene ring, which may be superimposed in six different ways by virtually rotating the query substructure.

Results

For the evaluation of the CFM based method of subgraph/ substructure search we used a test data set containing

Table 2 Three different sets of SMARTS patterns used with the PATTY algorithm

Substructure	SMARTS pattern
	Pattern set A "Chemical element"
Ethanol	C~C~O
Acetic acid	C~C(~O)~O
Urea	N~C(~O)~N
Glycine	N~C~C(~O)~O
Pyrrole	c1cncc1
Indole	c12ccccc1ccn2
	Pattern set B "Chemical element and binding type"
Ethanol	[CQ1X4]~[CQ2X4]~[OQ1X2]
Acetic acid	[CQ1X4]~[CQ3X3](~[OQ1X1])~[OQ1X2]
Urea	[NQ1X3]~[CQ3X3](~[OQ1X1])~[NQ1X3]
Glycine	[NQ1X3]~[CQ2X4]~[CQ3X3](~[OQ1X1])~[OQ1X2]
Pyrrole	c1cncc1
Indole	c12ccccc1ccn2
	Pattern set C "Biological property"
Ethanol	[CH3,cH3,CQ1H2,cQ1H2,CQ1H1,cQ1H1]~*~[$([NH2]-A),$([OH]-*);!+;!-]
Acetic acid	[CH3,cH3,CQ1H2,cQ1H2,CQ1H1,cQ1H1]~*(~[OQ1X1,oQ1X1,OQ2-X2,oQ2X2,NQ3X3,nQ3X3,NQ2X2,nQ2X2,NQ1X1,nQ1X1])~[$([NH2]-A),$([OH]-*);!+;!-]
Urea	[$([NH2]-A),$([OH]-*);!+;!-]~*(~[OQ1X1,oQ1X1,OQ2X2,oQ2X2,NQ3X3,nQ3X3,NQ2X2,nQ2X2,NQ1X1,nQ1X1;!+;!-])~[$([NH2]-A),$([OH]-*);!+;!-]
Glycine	*(~[OQ1X1,oQ1X1,OQ2X2,oQ2X2,NQ3X3,nQ3X3,NQ2X2,nQ2X2,NQ1X1,nQ1X1])(~[$([NH2]-A),$([OH]-*);!+;!-])~*~[$([NH2]-A),$([OH]-*);!+;!-]
Pyrrole	a1a[$([NH2]-a),NQ1H3,NQ2H2,NQ3H1,NQ2H1,nQ1H3,nQ2H2,nQ3H1, nQ2H1,$(Cl-*),$(Br-*),$(I-*);!+;!-]aa1
Indole	a12aaaaa1aa[$([NH2]-a),NQ1H3,NQ2H2,NQ3H1,NQ2H1,nQ1H3,nQ2H2,nQ3H1,nQ2H1,$(Cl-*),$(Br-*),$(I-*);!+]2

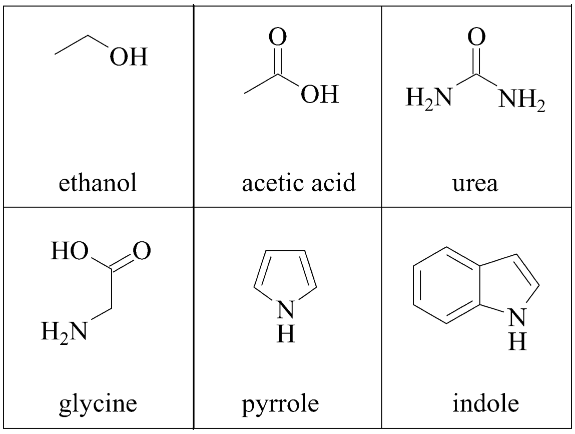

Fig. 6 Chemical structures of the six query substructures ethanol, acetic acid, urea, glycine, pyrrole and indole

100,000 molecules that were selected from the first 100,022 compounds of a freely available database ("jan02_2d") provided by the National Cancer Institute, Bethesda, Md. [21] Due to an incorrect ring assignment by the SSSR method, 22 of the first 100,022 compounds were left out. Problems concerning the detection of an optimum ring set have been reported. [12] These compounds were searched for the six substructures ethanol, acetic acid, urea, glycine, pyrrole and indole (Fig. 6) using both the CFM based search method and the graph algorithm PATTY.

To evaluate the effect of varying kinds of structural representation on the search results and on computing time, the CFM method was performed with the three different feature sets shown in Table 1. The particular feature types are represented by means of SMARTS strings.

Thereby, feature set "A" merely distinguishes between aliphatic and aromatic occurrences of the three elements carbon, oxygen and nitrogen. In addition to this, feature set "B" contains further information about binding types. Within feature set "C", different structures that exhibit similar biological properties are grouped and assigned to the respective feature. As an example, in the first SMARTS string of this set, "H" stands for implicit and explicit hydrogens while "Q" represents each kind of nonhydrogen atom. Accordingly, the three patterns stand for the three types of terminal carbon atoms, $–CH_3$, $=CH_2$ and $\equiv CH$, respectively. For a detailed description of SMARTS see [8]. According to the described feature sets, the PATTY algorithm was also run three times for each substructure. The respective sets of SMARTS patterns are displayed in Table 2.

Table 3 shows the search results as well as the computing times of the described evaluations, grouped by the focus of the structural representations, i.e. "chemical element", "chemical element and binding type" and "biological property".

The second column of Table 3 shows the number of compounds that were found to include the searched substructure, depending on the respective type of representation. Therein, only those compounds are regarded

Table 3 Search results of the PATTY algorithm and the CFM method [without and with preselection (PS)]. The three parts of the table display the three different structural representations. The test data set used contains 100,000 compounds

	Compounds found	Evaluation time (s)		
		PATTY	CFM	CFM (PS)
Focus: chemical element				
Ethanol	54,901	425	13	11
Acetic acid	20,920	410	16	12
Urea	3,337	403	10	3
Glycine	3,690	402	15	4
Pyrrole	2,022	401	28	10
Indole	1,303	417	248	124
Focus: chemical element and binding type				
Ethanol	29	400	3	1
Acetic acid	108	400	3	1
Urea	4	397	3	1
Glycine	7	399	5	1
Pyrrole	2,022	400	28	11
Indole	1,303	418	247	123
Focus: biological property				
Ethanol	2,004	424	2	0.4
Acetic acid	173	424	3	0.6
Urea	1,039	414	3	0.8
Glycine	1,359	473	3	0.7
Pyrrole	4,035	454	1	0.4
Indole	2,917	419	2	0.4

that contain the substructure without any additional ring closures. For example, epoxides are neglected when searching for ethanol, even if it is merely represented by the succession of its chemical elements (feature set “A”). As expected, the more restrictive the structural representation is, the fewer compound are found.

The main difference between the two methods of substructure search—and thus the main advantage of the CFM—is the calculation time. Depending on the query structure and the particular feature set, the CFM method (even without preselection) is between two and several hundred times faster than the graph algorithm (Table 3, column 3 versus column 4). The largest differences in computing time are observed when complex structural patterns are applied, as in pattern/feature set “C”: On one hand, due to the logical disjunctions within the SMARTS patterns, the assignment of the searched structures is more costly than with the simpler pattern sets “A” and “B”. Accordingly, the PATTY algorithm is slowed down. On the other hand, using feature set “C”, the extensive preprocessing of molecular data results in very compact CFMs. To what extent the size of a CFM is influenced by the feature set may be viewed in Fig. 2. Since the speed of the CFM algorithm depends on the number of comparable matrix entries, small CFMs are superimposed much faster than large ones. This becomes obvious when the search times of indole, regarding feature sets “B” and “C”, are compared. In the first case, the row vector of the CFM consists of nine features. Thus, the upper triangular matrix of the corresponding 9×9 matrix contains 36 entries, which must all be matched to the CFM of each compared molecule. Note that (a) the CFMs of the tested compounds are also comparatively large (since they are calculated from the same feature set as the query structure) and (b) the query CFM may occur several times within the test CFM. Altogether, these factors result in a search time of about 4 min (which is still almost twice as fast as the PATTY algorithm). In contrast, the row vector of the CFM which is calculated from feature set “C” comprises only three features: DON, AR5 and AR6. Accordingly, its upper triangular matrix merely contains three entries, which obviously lowers the number of possible comparisons. Therefore, the required search time is reduced to 2 s. As mentioned above, since feature set “C” is based on biological properties instead of atoms, this kind of representation is most suitable regarding pharmacochemical applications.

As displayed in the last column of Table 3, the search time of the CFM algorithm may be further decreased by applying the two described methods of preselection. Obviously, this effect is the more significant, the fewer tested molecules contain the searched substructure.

In some special cases, a particular substructure is encoded by a single feature. For example, using feature set “C”, the substructure “benzene” is exclusively described by the feature “AR6”. In such a case, the matrix based search algorithm is bypassed and only the row vector of the CFM is screened for the occurrences of the given feature (resembling the common substructure descriptors). Accordingly, the search for “benzene”/“AR6” within the 100,000 tested compounds was performed in 120 ms.

Discussion

According to its generation, structure and utilizability, the CFM is to some degree an intermediate between graph algorithms and substructure descriptors. On one hand, the assignment of the user defined features to the respective structural patterns of the molecules requires a graph algorithm for substructure search. In this respect, the CFM resembles the above mentioned substructure descriptors. On the other hand, there are some striking advantages over the two methods: while common substructure descriptors are used for similarity and screening evaluations, they are not capable of a substructure/subgraph search themselves. That is, they do not display the overall topology or geometry of the described compounds. In contrast, the CFM descriptor allows both the screening for user defined features and the search for complex structures that are composed of these features. Compared to graph algorithms, the CFM descriptor enables a much faster search, which is due to the matrix based algorithm and to the preprocessing of the graph information. The latter is the main step of the conversion from (e.g.) the MDL Molfile format to the CFM format. As mentioned, for the test data set used this step took between 15 and 20 min, depending on the complexity of the selected features. But since an average similarity search with the PATTY graph algorithm lasts between 6 and 7 min, the CFM method already pays after the third run, presumed a suitable feature set. The very fast search times (between 0.1 and a few seconds for 100,000 compounds) of the described technique complies with the requirements of an interactive usage, even for data sets of up to about a million components. Furthermore, using the CFM format, the substructure search may be combined with the other methods implemented in the software COFEA: pharmacophore development and similarity search.

Currently, the CFM based substructure search is restricted to topological data. But since the CFM descriptor may also be built on the basis of Euclidean distances, further investigation will be done to extend the described algorithm to the application of geometrical data.

Acknowledgments This work was realized within the scope of the SOL project (Search and Optimization of Lead structures) which is supported by the German Federal Ministry of Education and Research, bmb+f under contract number 311681.

References

1. Todeschini R, Consonni V (2000) Handbook of molecular descriptors. Wiley-VCH, Weinheim, p 427
2. Ihlenfeld WD, Gasteiger J (1994) J Comput Chem 15:793–813
3. Hurst T, Heritage TW (1997) HQSAR. A highly predictive QSAR technique based on molecular holograms. In: 213th ACS National Meeting, San Francisco, Calif.
4. Seel M, Turner DB, Willett P (1999) Quant Struct Act Relat 18:245–252
5. Carhart RE, Smith DH, Venkataraghavan R (1985) J Chem Inf Comput Sci 25:64–73
6. Scsibrany H, Varmuza K (1992) Topological similarity of molecules based on maximum common substructures. In: Ziessow D (ed) Software development in chemistry. Proceedings of the 7th CIC Workshop “Computers in Chemistry”, Berlin
7. Ullmann JR (1976) J Assoc Comput Mach 23:31–42
8. Daylight Chemical Information Systems (2002) Daylight theory manual, http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html
9. Rücker G, Rücker C (2001) J Chem Inf Comput Sci 33:1457–1462
10. Figueras J (1996) J Chem Inf Comput Sci 36:986–991
11. Fujita S (1988) J Chem Inf Comput Sci 28:1–9
12. Downs GM, Gillet VJ, Holliday JD, Lynch MF (1989) J Chem Inf Comput Sci 29:187–206
13. Dury L, Latour T, Leherte L, Barberis F, Vercauteren DB (2001) J Chem Inf Comput Sci 41:1437–1445
14. Abolmaali SFB, Ostermann C, Zell A (2003) J Mol Model, in press
15. Bush BL, Sheridan RP (1993) J Chem Inf Comput Sci 33:756–762
16. Wegner JK, Zell A (2002) JOELib—a java based computational chemistry package. 16th Molecular Modeling Workshop, Darmstadt
17. JOELib (2002) http://sourceforge.net/projects/joelib
18. Böhm M, Klebe G (2002) J Med Chem 45:1585–1597
19. MDL Information Systems (2002) CTfile formats, http://www.mdli.com/downloads/literature/ctfile.pdf
20. Dalby A, Nourse JG, Hounshell WG, Gushurst AKI, Grier DL, Leland BA, Laufer J (1992) J Chem Inf Comput Sci 32:244–255
21. National Cancer Institute, Bethesda, Md., http://dtp.nci.nih.gov/webdata.html