Acta Univ. M. Belii Math.
No. 11(2004), pp. 27–44

# USING A COMPUTER IN MATROID THEORY RESEARCH

Petr Hliněný

Abstract. In this paper we introduce our computer program Macek for structural computations with matroids representable over finite (partial) fields.
See http://www.mcs.vuw.ac.nz/research/macek. Using this program, we then find all 56 ternary excluded minors for the class of matroids of branch-width three. That research continues on the binary case [P. Hliněný, *On the Excluded Minors for Matroids of Branch-Width Three*, Electronic Journal of Combinatorics 9 (2002), #R32].

## 1 Introduction

Matroids represented over a finite (partial) field play an important role in structural matroid theory, similar to the role that graphs embedded on a surface play in structural graph theory. However, unlike for embedded graphs, it is difficult to visualize a matroid in rank bigger than 3, even when it is given as a matrix or a vector configuration. It is even more difficult to examine basic structural properties of given matroids like isomorphism, minors, connectivity, branch-width, or matroid extensions.

It is often the case that proving a theorem in structural matroid theory requires one to check all the small cases (on about, say, 10 elements) by hand, or to verify specific properties of selected small matroids, which are often represented by matrices over finite fields. In graph theory, such tasks are easily solved with a pen and a paper, but, unfortunately, it is not like that with matroids. As matroid researchers know very well themselves, checking the "small cases" can be quite long and painful, and prone to errors. Such is the situation with the problem of finding the excluded minors for matroids of branch-width three we focus on here – it is known that the excluded minors have at most 14 elements.

That is why we have developed a computer program Macek [6] for practical structural computations with matroids represented over finite partial fields. This program supports an easy manipulation and computations with matrices representing matroids over finite partial fields. For example, one can test for matroid minors, equivalence, representability, isomorphism, branch-width three, connectivity, and other structural properties. An important function is an exhaustive generation of all 3-connected extensions of matroids. The program is free, distributed under

*2000 Mathematics Subject Classification.* Primary 05B35; Secondary 68R05.
*Key words and phrases.* matroid, matroid extension, exhaustive generation, branch-width.
Partial support by Slovak grant VEGA #1/1002/04.
Received 13. 5. 2004; Accepted 21. 9. 2004

the terms of the GNU General Public License as published by the Free Software Foundation. See [6] for information about how to obtain and install the MACEK program.

## 2 Basics of Matroids

We refer to Oxley [10] for matroid terminology. A *matroid* is a pair $M = (E, \mathcal{B})$ where $E = E(M)$ is the ground set of $M$ (elements of $M$), and $\mathcal{B} \subseteq 2^E$ is a nonempty collection of *bases* of $M$. Moreover, matroid bases satisfy the "exchange axiom"; if $B_1, B_2 \in \mathcal{B}$ and $x \in B_1 - B_2$, then there is $y \in B_2 - B_1$ such that $(B_1 - \{x\}) \cup \{y\} \in \mathcal{B}$. We consider only finite matroids. Subsets of bases are called *independent sets*, and the remaining sets are *dependent*. Minimal dependent sets are called *circuits*. All bases have the same cardinality called the *rank* $\mathrm{r}(M)$ of the matroid. The *rank function* $\mathrm{r}_M(X)$ in $M$ is the maximal cardinality of an independent subset of a set $X \subseteq E(M)$.

If $G$ is a (multi)graph, then its *cycle matroid* on the ground set $E(G)$ is denoted by $M(G)$. The independent sets of $M(G)$ are acyclic subsets (forests) in $G$, and the circuits of $M(G)$ are the cycles in $G$. Another example of a matroid is a finite set of vectors with usual linear dependency. If $\mathbf{A}$ is a matrix, then the matroid formed by the column vectors of $\mathbf{A}$ is called the *vector matroid* of $\mathbf{A}$.

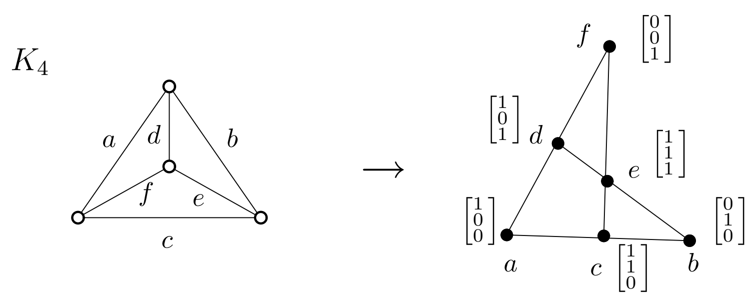


**Fig. 1**. An example of a vector representation of the cycle matroid $M(K_4)$. The matroid elements are depicted by dots, and their (linear) dependency is shown using lines.

The *dual* matroid $M^*$ of $M$ is defined on the same ground set $E$, and the bases of $M^*$ are the set-complements of the bases of $M$. A set $X$ is *coindependent* in $M$ if it is independent in $M^*$. An element $e$ of $M$ is called a *loop* (a *coloop*), if $\{e\}$ is dependent in $M$ (in $M^*$). The matroid $M \setminus e$ obtained by *deleting* a non-coloop element $e$ is defined as $(E - \{e\}, \mathcal{B}^-)$ where $\mathcal{B}^- = \{B : B \in \mathcal{B}, e \notin B\}$. The matroid $M/e$ obtained by *contracting* a non-loop element $e$ is defined using duality $M/e = (M^* \setminus e)^*$. (This corresponds to contracting an edge in a graph.) Conversely, a matroid $M'$ is a one-element *extension* (*coextension*) of $M$ if $M = M' \setminus e$ ($M = M'/e$) for some element $e$. A *minor* of a matroid is obtained by a sequence of deletions and contractions of elements. Since these operations naturally commute, a minor $M'$ of a matroid $M$ can be uniquely expressed as $M' = M \setminus D/C$ where $D$ are the coindependent deleted elements and $C$ are the independent contracted elements.

### Matroid Connectivity

An important concept in structural matroid theory is *connectivity*, which is close, but somehow different, to traditional graph connectivity. The *connectivity function* $\lambda_M$ of a matroid $M$ is defined for all subsets $A \subseteq E$ by

$$\lambda_M(A) = r_M(A) + r_M(E - A) - r(M) + 1 \,.$$

Here $r(M) = r_M(E)$. A subset $A \subseteq E$ is *k-separating* if $\lambda_M(A) \leq k$. A partition $(A, E - A)$ is called a *k-separation* if $A$ is $k$-separating and both $|A|, |E - A| \geq k$. Geometrically, the spans of the two sides of a $k$-separation intersect in a subspace of rank less than $k$. See in Fig. 2. In a corresponding graph view, the connectivity function $\lambda_G(F)$ of an edge subset $F \subseteq E(G)$ equals the number of vertices of $G$ incident both with $F$ and with $E(G) - F$. (Then $\lambda_G(F) = \lambda_{M(G)}(F)$ provided both sides of the separation are connected in $G$.) For $n > 1$, a matroid $M$ is *n-connected* if it has no $k$-separation for $k = 1, 2, \ldots, n - 1$, and $|E(M)| \geq 2n - 2$.

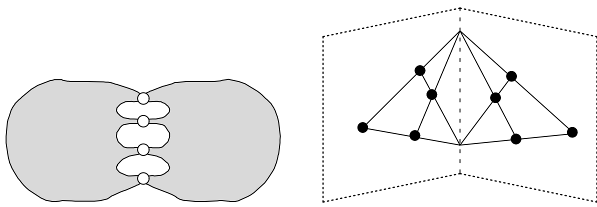

**Fig. 2**. An illustration to a 4-separation in a graph, and to a 3-separation in a matroid.

Of particular interest to us are 3-connected matroids, which capture the core of most structural properties and problems on matroids. 3-connected matroids can be reasonably easily handled using so called Seymour's Splitter Theorem [14]. Let the *k-wheel* be the matroid $M(W_k)$ where $W_k$ is the graph obtained from a $k$-cycle by adding one vertex adjacent to all other vertices. The *k-whirl* is obtained from the $k$-wheel by relaxing (making independent) the rim circuit.

**Theorem 1.** (Seymour) Let $M, N$ be 3-connected matroids such that $N$ is a minor of $M$. Suppose that if $N$ is a wheel (a whirl), then $M$ has no larger wheel (no larger whirl) as a minor. Then there is a 3-connected matroid $N_1$ such that $|E(N_1)| = |E(N)| + 1$, and that $M$ has an $N_1$-minor.

This important theorem allows a step-by-step construction of large 3-connected matroids from smaller ones; adding only one element at each step while maintaining 3-connectivity. (In other words, doing 3-connected one-element extensions and coextensions.)

**Matroid Representations**

An $\mathbb{F}$-*representation* of a matroid $M$ is a matrix $\mathbf{A}$ over a field $\mathbb{F}$ whose columns correspond to the elements of $M$, and linearly independent subsets of columns form the independent sets of $M$. Alternatively, one may view the matrix $\mathbf{A}$ as a point configuration in a projective space over $\mathbb{F}$. A matrix $\mathbf{A}$ is in the *standard form* if the number of rows in $\mathbf{A}$ equals the rank of $M$, and if some basis of $M$ is displayed in $\mathbf{A}$ as a unit submatrix. A matrix $\mathbf{A}'$ is a *reduced representation* of the matroid $M = M(\mathbf{A}')$ if $[\mathbf{I} \,|\, \mathbf{A}']$ is the standard form matrix representing $M$.

Moreover, we consider matroids represented over partial fields. A *partial field* is a generalization of a field, in which the addition is a partial operation. We refer to [13] for a formal definition and properties of partial fields. A typical and well-known example is the *regular* partial field consisting of the integers $-1, 0, 1$ with usual addition and multiplication. A matrix $\mathbf{A}$ over a partial field $\mathbb{P}$ is *proper* if all subdeterminants of $\mathbf{A}$ are defined in $\mathbb{P}$. For example, proper regular matrices are traditionally known as totally-unimodular. A matroid $N$ is representable over $\mathbb{P}$ iff there is a proper matrix $\mathbf{A}$ over $\mathbb{P}$ such that $N \simeq M(\mathbf{A})$.

A partial field is called *finite* if the equation $x-1=y$ has finitely many solutions in $\mathbb{P}$. All finite fields are clearly finite in this sense. However, a finite partial field may have infinitely many elements. (The reason for our terminology is that a fixed-rank simple matroid representable over a finite partial field may have only finite number of elements.)

We say that two matrices are *strongly equivalent* if one can be obtained from the other by a sequence of row or column permutations, non-zero scalings, and pivots. Considering reduced $\mathbb{P}$-representations of matroids, we call the $\mathbb{P}$*-represented matroid* an equivalence class of unlabeled matrices with respect to the strong equivalence. Clearly, represented matroids refine the isomorphism classes of matroids. On the other hand, one matroid may have several non-equivalent representations over $\mathbb{P}$. An obvious example of this phenomenon is presented in Fig. figdifrep.

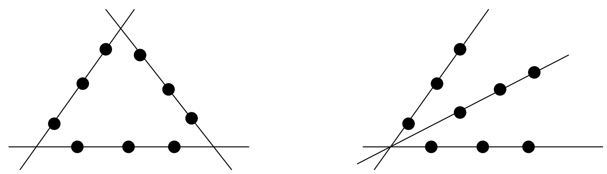

**Fig. 3**. Two inequivalent representations of a 9-element rank-3 matroid.

A matroid $M$ is *regular* if $M$ is representable over the regular partial field. A regular matroid is then representable over all fields. A matroid $M$ is *binary, ternary*, if $M$ is representable over the fields $GF(2)$, $GF(3)$, respectively. We remark that cycle matroids of graphs are regular.

## Small matroid enumeration

To introduce and demonstrate capabilities of the Macek program in practice, we first present a table summarizing enumeration of small 3-connected regular, binary, ternary, and quaternary matroids. Let $U_{2,k}$ denote the rank-2 matroid formed by $k$ distinct points in one line.

| representable \ elements | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| regular | 0 | 0 | 1 | 0 | 1 | 4 | 7 | 10 | 33 | 84 | 260 | 908 |
| $GF(2)$, non-regular | 0 | 0 | 0 | 2 | 2 | 4 | 17 | 70 | 337 | 2080 | 16739 | 181834 |
| $GF(3)$, non-regular | 1 | 0 | 1 | 6 | 23 | 120 | 1045 | 14116 | 330470 | ? | ? | ? |
| $GF(4)$, non-$GF(2,3)$ | 0 | 2 | 2 | 8 | 69 | 748 | 15305 | ? | ? | ? | ? | ? |

Tab. 1. The numbers of 3-connected matroids representable over small fields.

The results in Table 1 have been obtained using Theorem 1 and the following Macek computations:

- The numbers in the first row have been computed via regular 3-connected extensions of all small wheels (removing duplicities).

– The numbers in the second row have been obtained using binary 3-connected extensions of the Fano matroid (the binary projective plane), which is the smallest binary non-regular matroid.
– In the third row, we have computed all ternary 3-connected extensions of the 3-whirl matroid, and added the larger whirls. (Although every non-binary matroid contains the 4-point line $U_{2,4}$ as a minor, that matroid has no ternary extensions, and so we had to use a detour in our computation.)
– In the fourth row, we have computed all quaternary 3-connected extensions of the 5-point line $U_{2,5}$, which is the smallest non-ternary matroid. Moreover, we have removed isomorphic pairs of matroids afterward, as quaternary matroids already may have non-equivalent representations.

## 3 Overview of Macek Capabilities

We give a brief overview of our Macek program in this section. As we have already mentioned, the program has been developed to assist matroid theory research with useful structural computations. It is designed in a command-line oriented form, which is suitable especially for large-scale batch computations, but answering (single) structural questions is also supported well. We refer to [6] for a full description and technical details (including an installation instructions).

### Matrix representations

The Macek program deals with $\mathbb{P}$-represented matroids (given by reduced matrix representations) in the sense of the definition from Section 2. $\mathbb{P}$ may be an arbitrary finite partial field. Definitions of common small fields and partial fields are compiled in Macek, and it is not difficult to add other partial fields via description of generators of the multiplicative subgroup. Representations of many well-known matroids are also distributed with the program.

Since the basic entity in Macek is a $\mathbb{P}$-represented matroid – an equivalence class of matrices over $\mathbb{P}$, two non-equivalent matrices are considered distinct even if they represent isomorphic matroids. So the issue of inequivalent matroid representations has to be considered when it comes up, i.e. over fields larger than $GF(3)$. In this context, it is important to mention that matroid elements in Macek are not explicitly labeled (though they get implicit labels for the purpose of display). So an "equivalence" is meant to be the strong unlabeled equivalence of matrices.

### Structural functions

It is possible to compute various matroid tasks and properties with Macek: Those include looking for specific minors in given matroids, finding an equivalence or an abstract isomorphism between matroids, computing matroid connectivity or girth (shortest cycle length), etc. Other specific functions test for branch-width three or for paving matroids, etc. All these functions can also be applied as filters to (generated) matroid lists.

We remark that such structural properties are usually computationally very hard, and hence we have to implement most of them using clever adaptations of brute-force mathods. The bad side is that computational time grows exponentially, and usually only matroids on less than 20 elements could be efficiently handled. Still, the program functions seem to be enough powerful and fast to substantially help with matroid theory research.

Besides those, Macek can compute and print out various structural information about a matroid itself, like bases, automorphism group orbits, small flats and separations, connectivity, and representability over other fields. For example, the following extensive information can be printed about the matroid $R_{12}$. ($R_{12}$ is an interesting matroid playing a crucial role in Seymour's decomposition theorem [14] for regular matroids.)

```
  MACEK 1.1.9999 (23/04/04) starting...

vv=====================================================================================vv

~532~          Output of the command "!prmore ((t))   [1]":

    ~   -------------------------------------------------------------

    ~   matrix 0x8190168 [R12], r=6, c=6, tr=0, ref=(nil)

    ~            '-1')   '-2')   '-3')   '-4')   '-5')   '-6')

    ~    '1')       1       1       1       o       o       o

    ~    '2')       1       1       o       1       o       o

    ~    '3')       1       o       o       o       1       o

    ~    '4')       o       1       o       o       o       1

    ~    '5')       o       o       1       o      -1      -1

    ~    '6')       o       o       o       1      -1      -1

    ~   -------------------------------------------------------------

~532~   Number of matroid [R12] bases:  441

~532~     - per elements  [1: 210] [2: 210] [3: 231] [4: 231] [5: 210] [6:210]

                       [-1: 231] [-2: 231] [-3: 210] [-4: 210] [-5: 231] [-6: 231]

~532~   Automorphism group orbits of [R12] are (via first elem id):

                (1, 1, 3, 3, 1, 1,  3, 3, 1, 1, 3, 3)

~535~   There are -NO- (nontrivial) flats in [R12] of rank 0.

~535~   There are -NO- (nontrivial) flats in [R12] of rank 1.

~535~   Listing all (nontrivial) flats in [R12] of rank 2:

    ~    - rank-2 flat (1)      { 1, 5,  -3 }

    ~    - rank-2 flat (2)      { 2, 6,  -4 }

~535~   Listing all (nontrivial) flats in [R12] of rank 3:

                ......... <skipped> .........

~535~   There are -NO- exact separations in [R12] of lambda 1.

~535~   There are -NO- exact separations in [R12] of lambda 2.

~535~   Listing all exact separations in [R12] of lambda 3:

    ~    - 3-separation (1)     ( 1, 2, 5, 6,  -3, -4, )

    ~    - 3-separation (2)     ( 1, 5,  -3, )

    ~    - 3-separation (3)     ( 2, 6,  -4, )

    ~    - 3-separation (4)     ( 3,  -1, -5, )

    ~    - 3-separation (5)     ( 3, 4,  -1, -2, -5, -6, )

    ~    - 3-separation (6)     ( 4,  -2, -6, )

~535~   Matroid [R12] connectivity is 3.

~535~   Matroid [R12] girth (shortest cycle) is 3.

~535~   Matroid [R12] representability:

         +GF(2)+ +GF(3)+ +GF(4)+ +GF(5)+ +GF(7)+ +GF(8)+ +GF(9)+

^^=====================================================================================^^
```

### Matroid generation

In order to use a computer in proving general statements about matroids, we need a suitable tool for exhaustive generation of matroids. Due to the existence of enormous numbers of matroids already on a few elements, MACEK supports generating matroid extensions rather than generating from scratch. This approach seems to be better suited for practical applications. A theoretical description of the (quite involved) generation algorithm used in MACEK is presented in [8]. Our algorithm allows for a multi-step equivalence-free generation of extensions, which can be, moreover, easily distributed in a parallel computing environment without need for inter-process communication.

Likewise, one can ask MACEK to generate all nonequivalent single-element 3-connected extensions and coextensions of the matroid $R_{10}$ which are representable over $GF(5)$. The answer is as follows,

```
sh$ macek -pgf5 '!extend' R10
      MACEK 1.1.9999 (23/04/04) starting...
    ~979~   Generated 12 non-equiv 3-conn row co-extensions of the sequence [R10] (5x5|5x5).
    ~985~   Generated 12 non-equiv 3-conn column extensions of the sequence [R10] (5x5|5x5).
    ~985~   In total 24 (co-)extensions of 1 matrix-sequences generated for "b" over GF(5).
```

and the 24 generated extensions can be readily used in further computations. For example, a subsequent test can find out that two of the 12 coextensions there have girth 5, i.e. they have no circuits on less than 5 elements. Or, that all generated extensions are pairwise non-isomorphic here. The multi-step feature of our generation algorithm allows to start next steps independently from each of the previous extensions, and yet to generate no duplicated extensions.

The above mentioned matroid $R_{10}$ is well known for being a splitter for the class of regular matroids. (A *splitter* has no 3-connected extension or coextension in its class.) Using MACEK, one can easily prove that $R_{10}$ is a splitter also for the class of all near-regular matroids (those representable over all fields larger than $GF(2)$, at least).

```
sh$ macek -pnreg '!extend' R10
  MACEK 1.1.9999 (23/04/04) starting...
~126~   Generated 0 non-equiv 3-conn row co-extensions of the sequence [R10] (5x5|5x5).
~126~   Generated 0 non-equiv 3-conn column extensions of the sequence [R10] (5x5|5x5).
~126~   In total 0 (co-)extensions of 1 matrix-sequences generated for "b" over near-reg.
```

Besides matroid extensions, MACEK supports generation of all representations of a matroid over a given field. (That, of course, includes simply testing representability over a field.) For example, one may find out that the uniform matroid $U_{3,6}$ has 140 representations over the field $GF(7)$ that are distinct up to scaling, but only three of them are inequivalent (in the unlabeled sense). As another example, one may compute that each of the 6 single-element extensions of $U_{3,6}$ over $GF(7)$ have more than one pairwise non-equivalent representations there:

```
sh$ macek -pgf7 '!verbose;!extend c;!represgen "" allq' U36
  MACEK 1.1.9999 (23/04/04) starting...
~333~   Generated 6 non-equiv 3-conn column extensions of the sequence [U36] (3x3|3x3).
~333~   In total 6 (co-)extensions of 1 matrix-sequences generated for "c" over GF(7).
~334~   There are 2 nonequiv GF(7)-representations of #1 matroid [U36_c1] (3x4, GF(7)).
```

```
~334~   There are 10 nonequiv GF(7)-representations of #2 matroid [U36_c2] (3x4, GF(7)).
~335~   There are 4 nonequiv GF(7)-representations of #3 matroid [U36_c3] (3x4, GF(7)).
~335~   There are 10 nonequiv GF(7)-representations of #4 matroid [U36_c4] (3x4, GF(7)).
~336~   There are 10 nonequiv GF(7)-representations of #5 matroid [U36_c5] (3x4, GF(7)).
~337~   There are 2 nonequiv GF(7)-representations of #6 matroid [U36_c6] (3x4, GF(7)).
```

### Other capabilities

Lastly, we briefly mention other supplementary functions in Macek. Those include mainly file reading and writing operations, and operations manipulating with single matrices and with whole lists of them. (Actually, all data in Macek are structured in a tree-like fashion.) Moreover, Macek offers basic scripting capabilities like procedures, conditions, jumps, and others. We refer to the manual for a full description, and for many examples of use.

## 4 Excluded Minors for Branch-Width Three

We now move to the main topic of research in this paper. The concept of graph tree-width is rather well known nowadays. A similar, but less known, structural parameter is called branch-width, and it is within a constant factor of tree-width on graphs.

Let $\lambda$ be a symmetric function on the subsets of a ground set $E$. (Here $\lambda \equiv \lambda_G$ is the connectivity function of a graph, or $\lambda \equiv \lambda_M$ of a matroid.) A *branch decomposition* of $\lambda$ is a pair $(T, \tau)$ where $T$ is a sub-cubic tree ($\Delta(T) \leq 3$), and $\tau$ is a bijection of $E$ into the leaves of $T$. For $e$ being an edge of $T$, the *width* of $e$ in $(T, \tau)$ equals $\lambda(A) = \lambda(E - A)$, where $A \subseteq E$ are the elements mapped by $\tau$ to leaves of one of the two connected components of $T - e$. The width of the branch decomposition $(T, \tau)$ is maximum of the widths of all edges of $T$, and *branch-width* of $\lambda$ is the minimal width over all branch decompositions of $\lambda$.

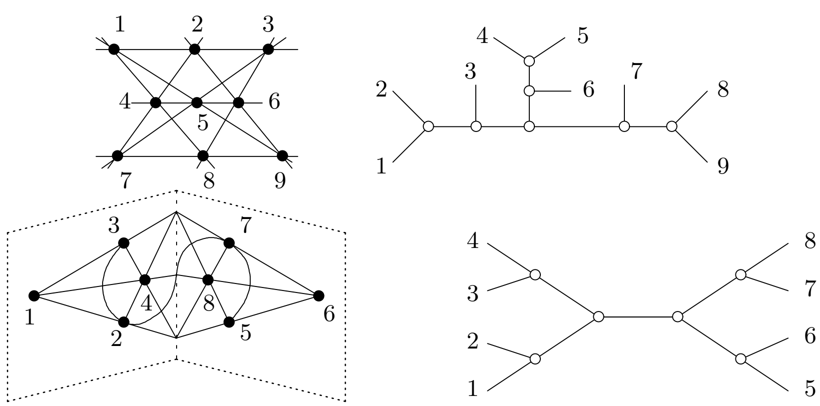

**Fig. 4**. Two examples of width-3 branch decompositions of the Pappus matroid (top left, in rank 3) and of the binary affine cube (bottom left, in rank 4). The lines in matroid pictures show dependencies among elements.

Recall the definitions of graph and matroid connectivity functions from Section 2. Then branch-width of $\lambda \equiv \lambda_G$ is called *branch-width of a graph G*, and that of $\lambda \equiv \lambda_M$ is called *branch-width of a matroid M*. (See examples in Fig. 4.) We remark that it is possible to define matroid tree-width [9] which is within a constant factor of branch-width, but this is not a straightforward extension of traditional graph tree-width.

The main focus of this research is on the class $\mathcal{B}_3$ of all matroids of branch-width at most three. Clearly, this class is minor-closed. A matroid $N$ is said to be an *excluded minor* for a minor-closed family $\mathcal{N}$ if $N \not\in \mathcal{N}$ but all proper minors of $N$ belong to $\mathcal{N}$. The question is: What are the excluded minors for the class $\mathcal{B}_3$? We base our (partial) answer to the question (further Theorems 4 and 5, Proposition ??? on the following theorem [4,5]:

**Theorem 2.** (Hall, Oxley, Semple, Whittle) If $N$ is an excluded minor for the class $\mathcal{B}_3$, then $N$ is a 3-connected matroid on at most 14 elements.

### The Binary Case

The story of the research originally started with considering the class of all graphs of branch-width at most three. All the excluded minors for this class were found first by Dharmatilake and others [3], but that research has not been publicized further. The same list was independently found later in [1].

**Theorem 3.** (Dharmatilake, Chopra, Johnson, Robertson) A graph has branch-width at most 3 if and only if it has no minor isomorphic to any one of the graphs $\{K_5, Q_3, O_6, V_8\}$. (See the graphs in Fig. 5.)

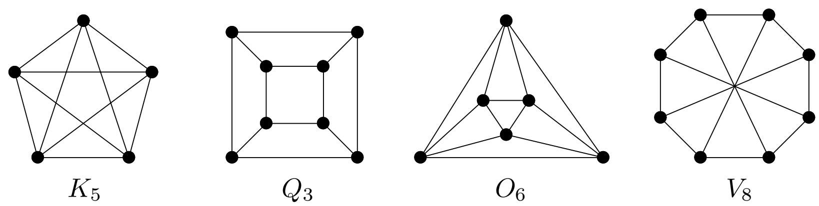


**Fig. 5**. The four excluded minors for graphs of branch-width at most 3.

It is easy to see that the cycle matroids of the graphs from Theorem 3 are also excluded minors for the matroid class $\mathcal{B}_3$. Moreover, the well-known regular matroid $R_{10}$ is an excluded minor for $\mathcal{B}_3$. Let us denote

$$\mathcal{R}_3 = \{M(K_5), M(K_5)^*, M(Q_3), M(O_6), M(V_8), M(V_8)^*, R_{10}\}\,.$$

Dharmatilake then used a specialized computer program to search for all small binary matroids (up to 12 elements) that are excluded minors for $\mathcal{B}_3$. He found three more non-regular matroids, denoted by $N_{11}, N_{23}, N_{11}^*$, and he conjectured [3] that $\mathcal{R}_3 \cup \{N_{11}, N_{23}, N_{11}^*\}$ is the complete set of binary excluded minors for $\mathcal{B}_3$.

$$R_{10} \quad \begin{bmatrix} -1 & 1 & 0 & 0 & 1 \\ 1 & -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 & 1 \\ 1 & 0 & 0 & 1 & -1 \end{bmatrix}$$

**Fig. 6**. The matroid $R_{10}$, in a totally unimodular (regular) representation.

We have finished [7] a computerized search of binary matroids up to 14 elements (cf. Theorem 2) using functions of our program Macek; which has turned out to be a much faster computation than Dharmatilake's one. (While Dharmatilake carried out a long computation in a supercomputing center, our search of binary matroids up to 12 elements took only a few seconds on a home PC computer. An extension up to 14 elements took another several hours.)

**Theorem 4.** (PH) A binary matroid has branch-width at most 3 if and only if it has no minor isomorphic to one of the members of $\mathcal{R}_3 \cup \{N_{11}, N_{23}, N_{11}^*\}$.

### The Ternary Case

A natural next step is to consider the same question – what are the excluded minors for the class $\mathcal{B}_3$, over ternary matroids, and so on. In theory, there is no problem with it, as an analogous Macek computation can be run over the field $GF(3)$. However, the complexity of computation grows enormously. That is illustrated in the next Table 2 showing the numbers of small regular, binary, and ternary members of $\mathcal{B}_3$.

| *representable \ elements* | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| regular | 0 | 0 | 1 | 0 | 1 | 4 | 4 | 8 | 23 | 46 | 123 |
| $GF(2)$, non-regular | 0 | 0 | 0 | 2 | 2 | 4 | 14 | 38 | 125 | 432 | 1551 |
| $GF(3)$, non-regular | 1 | 0 | 1 | 6 | 23 | 102 | 538 | 3008 | 18597 | 119594 | 796208 |

**Tab. 2**. The numbers of 3-connected matroids of branch-width three over small fields.

Yet we have been able to finish at least the search of ternary matroids up to 14 elements on a supercomputing cluster. Total computing time was equivalent to almost 2 years on a single 2GHz PC computer. Hence we have proved:

**Theorem 5.** There is a family $\mathcal{T}_3$ of 49 ternary non-regular matroids; such that a ternary matroid has branch-width at most 3 if and only if it has no minor isomorphic to one of the 56 members of $\mathcal{R}_3 \cup \mathcal{T}_3$.

*Proof.* By [15], all non-binary matroids contain a $U_{2,4}$-minor. Unfortunately, $U_{2,4}$ (isomorphic to the 2-whirl) is one of the exceptions in Theorem 1, but an enhancement of this theorem [2] (also in [[]Section 11.3]10) implies that all 3-connected ternary extensions of $U_{2,4}$, that are not whirls, contain a single-element extension or coextension of the 3-whirl $\mathcal{W}^3$ as a minor. All whirls clearly have branch-width three.

Hence each excluded minor for our class $\mathcal{B}_3$ contains a single-element extension or coextension of $\mathcal{W}^3$ as a minor, and so Theorem 1 can be applied here. We proceed our computation along the following scheme:

(1) Start with the family $\mathcal{L}_6 = \{\mathcal{W}^3\}$, and $\mathcal{T}_3 = \emptyset$.
(2) For $i = 6, 7, \dots, 13$, compute a list $\mathcal{X}_{i+1}$ of all single-element extensions and coextensions of the matroids in $\mathcal{L}_i$.
(3) Set $\mathcal{L}_{i+1}$ to be the set of all matroids from $\mathcal{X}_{i+1}$ that have branch-width at most three.
(4) Remove all matroids from $\mathcal{X}_{i+1} - \mathcal{L}_{i+1}$ that have minors in the current set $\mathcal{T}_3$ or in $\mathcal{R}_3$. Add the remaining matroids to the list $\mathcal{T}_3$.
(5) If $i < 14$, then go to 2 with $i + 1$.

After the first iteration of the scheme, $\mathcal{L}_7$ contains all 6 single-element extensions and coextensions of $\mathcal{W}^3$, all of them having branch-width three. Then every ternary excluded minor $X$ for the class $\mathcal{B}_3$ is eventually constructed in step 2 for $i \in \{8, \ldots, 13\}$, as that follows from Theorems 1 and 2. Such excluded minors $X$ are then identified in step 4, and stored in the list $\mathcal{T}_3$. On the other hand, every matroid $X \in \mathcal{T}_3$ has branch-width larger than three, and $X$ has no proper minor of branch-width more than three. So $X$ is an excluded minor for $\mathcal{B}_3$.

**Future work**

We have run the same procedure as described in the proof of Theorem 5 over other small fields $GF(4)$, $GF(5)$ and $GF(7)$. The only difference is that we have started the generating procedure from the list $\mathcal{L}_5 = \{U_{2,5}, U_{3,5}\}$, referring the result of [12]:

Any 3-connected non-binary non-ternary matroid representable over some field has a $U_{2,5}$- or $U_{3,5}$-minor.

Moreover, keeping in mind that matroids may have nonequivalent representations over fields larger than $GF(3)$, we have removed isomorphic pairs of matroids from the resulting lists. We present a summary of the results that we have obtained in the next Table 3.

| *representable \ elements* | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|
| regular | 0 | 0 | 0 | 3 | 0 | 4 | 0 | 0 |
| $GF(2)$, non-regular | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 |
| $GF(3)$, non-regular | 0 | 0 | 18 | 31 | 0 | 0 | 0 | 0 |
| $GF(4)$, non-$GF(2,3)$ | 0 | 5 | 90 | 32 | 0 | ? | ? | ? |
| $GF(5)$, non-$GF(2,3,4)$ | 0 | 38 | 444 | 29 | ? | ? | ? | ? |
| $GF(7)$, non-$GF(2,3,4,5)$ | 2 | 119 | 344 | ? | ? | ? | ? | ? |
| $GF(8)$, non-$GF(2,3,4,5,7)$ | 0 | 5 | ? | ? | ? | ? | ? | ? |
| $GF(9)$, non-$GF(2,3,4,5,7,8)$ | 0 | 0 | ? | ? | ? | ? | ? | ? |

**Tab. 3**. The numbers of excluded minors for matroids of branch-width three.

Notice, in particular, how many (small) excluded minors for the class $\mathcal{B}_3$ are there. This shows that the matroid class $\mathcal{B}_3$ has a quite rich structure, unlike its graphic counterpart which has only 4 excluded minors (Theorem 3).

**Proposition 6.** There are at least 1167 pairwise non-isomorphic excluded minors for the class $\mathcal{B}_3$ of all matroids of branch-width at most three.

As one can see in Table 3, we have not been able to finish the exhausted search up to 14 elements. We would better say that we have hit really hard the "wall of intractability" here. It appears that the number of the members of $\mathcal{B}_3$ up to 14 elements (regardless of representability) grows enormously, and hence it is simply impossible to finish the search for the excluded minors for $\mathcal{B}_3$ in general, even if one tried to design much faster algorithms than we have used here. However, the numbers in Table 3 still give a hope of finishing up the whole problem – it looks likely that there are no more "large" excluded minors for $\mathcal{B}_3$ than we already know.

Supported by our computing results, we propose the following strengthening of Theorem 2:

**Conjecture.** If $N$ is a non-regular excluded minor for the class $\mathcal{B}_3$ of all matroids of branch-width at most three, then $N$ has at most 10 elements.

Having a theoretical result like that at hand, it could be possible to carry out an exhaustive search of all abstract matroids on up to 10 elements, we think. (Unfortunately, the current version of Macek does not yet support computations with abstract, i.e. also non-representable, matroids.)

## 5 Reliability of Computations

A natural question a reader would probably ask here is: How reliable are the results of Macek computations? Computer-assisted proofs do not fit into the traditional scheme of mathematical proofs which could be verified step-by-step by hand, and so their wide acceptance could be sometimes controversial. (For example, look at the story of the famous "Four colour theorem".) However, everybody nowadays uses a calculator to do arithmetical operations, and nobody would doubt the results. Hence it is likely that a similar wide acceptance of computer-checked proofs will come soon.

In this section, we summarize the checks we have carried out to ensure that our computation results are correct. We divide the summary into two parts, one showing nontrivial internal relations between different parts of our computation, and the other one relating our computation results to other known research.

**Computing self-tests**

- All computations in Macek are backed by numerous internal self-checks, usually checking properties or relations, which follow from matroid theory but are not directly used in Macek algorithms. More details can be found in Macek source documentation.
- We have checked that the lists of excluded minors for $\mathcal{B}_3$ are closed under duality.
- The lists of all matroids of branch-width at most three over the (respective) fields $GF(2), GF(3)$ are obtained as side products in our computation. We have compared these lists with the lists independently computed via an enumeration of all small represented matroids (cf. Table 1), and selecting those members of branch-width at most three afterward. We have also verified that the intersection of the lists over $GF(2)$ and $GF(3)$ contains exactly the regular members.
- In the case of the fields $GF(4), GF(5), GF(7)$, over which a matroid may have inequivalent representations, we have checked that each isomorphism class of excluded minors generated in our computation really contains all possible inequivalent representations of it.
- We have also performed various "cross-representability" tests. That means, for $p, q \in \{4, 5, 7\}$, we have taken the lists $\mathcal{L}^p$, $\mathcal{L}^q$ of all generated matroids of branch-width three over $GF(p)$ and over $GF(q)$ (which are side products of our computation), and we have verified that the $GF(q)$-representable members of $\mathcal{L}^p$ match the $GF(p)$-representable members of $\mathcal{L}^q$. (Notice that such a test does not work with $p = 3$ since our procedure generates only non-ternary matroids over $GF(q)$ for $q > 3$.)

Of course, one may think about other self-tests that could be run with Macek, and the reader is welcome to download Macek [6] and try the tests.

**Comparing with other research**

- The binary excluded minors for the class $\mathcal{B}_3$ have been found by Dharmatilake [3]. So, first of all, we have compared our results [7] obtained over $GF(2)$ with that list. We remark that our computing approach to the problem has been quite different from Dharmatilake's approach.

- Moreover, a subsequent work of X. Zhou [16,17] has, among other interesting results, provided a hand-written proof of Theorem 4 (Dharmatilake's conjecture). Actually, his research has also been based on computations performed by MACEK, but then he has found clever arguments (based on the concept of internal 4-connectivity) that allowed him to narrow the exhausted search significantly, and so to write down all the steps and necessary arguments in a paper.
- Lastly we mention an exhaustive generation of matroids computed by R. Pendavingh [11], in a search for the excluded minors for matroids representable over $GF(5)$ and $GF(7)$. (We have run recently a similar computation, and the common parts of the results matched each other.) Although it is not directly related to our paper, we consider this recent feedback very important since it independently confirms correctness of our exhaustive generation process in MACEK over other fields than $GF(2)$.

## ACKNOWLEDGMENTS

The author, besides his current grant support, acknowledges generous support from the Victoria University of Wellington and from the New Zealand Marsden Fund during his stay in Wellington in 2000–2002, where development of the MACEK program has begun. The author also thanks Geoff Whittle for helpful ideas and comments in early stages of MACEK development.

The large-scale computations used in the proof of Theorem 5 were mostly run on the `termit` computing cluster at the Technical University Ostrava. Additional computing data summarized in Table 1 result from author's current computing project run on the `minos` cluster at the West Bohemia University (the ITI center, supported by the Ministry of Education of the Czech Republic as the project LN00A056).

## REFERENCES

[1] H. Bodlaender, D.M. Thilikos,, *Graphs with Branch-Width at most Three*, Technical Report UU-CS-1997-37, Department of Computer Science, Utrecht University, Utrecht, the Netherlands, 1997.

[2] C.R. Coullard,, *Minors of* 3*-Connected Matroids and Adjoints of Binary Matroids*, Ph.D. thesis, Northwestern University, 1985.

[3] J.S. Dharmatilake, *Binary Matroids of Branch-Width* 3, PhD. dissertation, Ohio State University, 1994.

[4] R. Hall, *Excluded Minors for the Matroids of Branch-Width Three*, M.Sc. Thesis, Victoria University of Wellington, 2000.

[5] R. Hall, J. Oxley, C. Semple, G. Whittle, *On Matroids of Branch-Width Three*, J. Combin. Theory Ser. B **86** (2002), 148–171.

[6] P. Hliněný,, *The* MACEK *Program*, `http://www.mcs.vuw.ac.nz/research/macek`, 2002–2004.

[6a] P. Hliněný,, *The* MACEK *Program*, `http://www.cs.vsb.cz./hlineny/MACEK`, 2002–2004.

[7] P. Hliněný,, *On the Excluded Minors for Matroids of Branch-Width Three*, Electronic Journal of Combinatorics **9** (2002), no. #R32.

[8] P. Hliněný, *Equivalence-Free Exhaustive Generation of Represented Matroids*, 2004 (submitted).

[9] P. Hliněný, G.P. Whittle,, *Matroid Tree-Width*, 2003, Extended abstract in: Eurocomb'03, ITI Series 2003–145, Charles University, Prague, Czech Republic, 202–205. (submitted).

[10] J.G. Oxley, *Matroid Theory*, Oxford University Press, 1992.

[11] R. Pendavingh, 2004, personal communication.

[12] C. Semple, G.P. Whittle, *On Representable Matroids Having Neither $U_{2,5}$-nor $U_{3,5}$-minors*, Matroid Theory, Contemporary Math. 197, Amer. Math. Soc., 1995, pp. 377–386.
[13] C. Semple, G.P. Whittle, *Partial Fields and Matroid Representation*, Advances in Appl. Math. **17** (1996), 184–208.
[14] P.D. Seymour, *Decomposition of Regular Matroids*, J. Combin. Theory Ser. B **28** (1980), 305–359.
[15] W.T. Tutte, *A Homotopy Theorem for Matroids*, Trans. Amer. Math. Soc. **88**, 144–174.
[16] X. Zhou, *Some Excluded Minor Theorems for Binary Matroids*, PhD. Thesis, The Ohio State University, 2003.
[17] X. Zhou, *On Internally* 4*-connected Non-regular Binary Matroids*, J. Combin. Theory Ser. B (2004) (to appear).

Department of Computer Science; VŠB – Technical University Ostrava; 17. listopadu 15, CZ–708 33 Ostrava; Czech Republic

E-mail: petr.hlineny@vsb.cz

Institute of Mathematics and Computer Science; Matej Bel University and Slovak Academy of Sciences; Severná 5, SK–974 00 Banská Bystrica ; Slovak Republic

E-mail: hlineny@savbb.sk

# Appendix A

Here we present the list $\mathcal{T}_3$ of all 49 non-regular ternary excluded minors for the class $\mathcal{B}_3$ of matroids of branch-width at most three. Each one is given as a reduced matrix representation over $GF(3)$.

```
1 0 1
1 1 0
0 1 1
0 1 2
1 0 2
1 1 1
1 2 0
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 0
1 0 2 2
0 1 0 2
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 0
1 0 2 2
0 0 1 1
```

```
1 0 1 0 0
1 1 0 0 0
0 1 1 1 1
0 1 2 0 2
1 0 2 2 0
```

```
1 0 1 0 0
1 1 0 0 0
0 1 1 1 1
0 1 2 0 1
1 0 2 2 0
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 0
0 1 2 0 2
1 0 2 2 0
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 0
0 1 2 0 2
1 0 2 2 2
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 1
0 1 2 2 0
1 0 2 2 2
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 1
0 1 2 2 1
1 0 2 2 2
```

```
1 0 1 0 1
1 1 0 0 1
0 1 1 1 2
0 1 2 2 0
1 0 2 2 2
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 0
1 1 1 1
0 0 1 1
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 0
1 1 1 1
0 1 2 1
```

```
1 0 1 0 0
1 1 0 0 1
0 1 1 1 0
0 1 2 0 1
1 1 1 1 1
```

```
1 0 1 0 0
1 1 0 0 1
0 1 1 1 2
0 1 2 0 1
1 1 1 1 1
```

```
1 0 1 0 0
1 1 0 1 1
0 1 1 1 2
0 1 2 1 2
1 1 1 2 2
```

```
1 0 1 0 0
1 1 0 0 1
0 1 1 1 0
0 1 2 1 1
0 1 0 1 0
```

```
1 0 1 0 0
1 1 0 0 1
0 1 1 1 2
0 1 2 1 1
0 1 0 1 2
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 1
0 0 1 1
1 0 1 2
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 0
0 1 2 1 2
0 0 1 1 0
```

```
1 0 1 0 1
1 1 0 0 2
0 1 1 1 0
0 1 2 1 1
0 0 1 1 0
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 1
0 1 1 0
1 1 0 1
```

```
1 0 1 0
1 1 0 0
0 1 1 1
1 1 1 1
0 0 1 1
0 1 0 1
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 0 0
0 1 2 1 1 2
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 0 0
0 1 2 1 1 1
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 2 0
0 1 2 1 1 2
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 2 0
0 1 2 1 1 1
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 1 2
0 1 2 1 0 0
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 0 2
0 1 2 1 2 2
```

```
1 0 1 0 1
1 1 0 0 0
0 1 1 1 0
1 1 1 1 1
0 0 1 1 1
```

```
1 0 1 0 0 1
1 1 0 0 1 0
0 1 1 1 0 0
1 1 1 1 1 1
```

```
1 0 1 0 1 1 1
1 1 0 1 0 1 2
0 1 1 2 2 1 0
```

```
1 0 1 0
1 1 0 0
0 1 1 0
0 1 2 1
1 0 2 2
```

```
1 0 1 0
1 1 0 0
0 1 1 0
0 1 2 1
1 0 2 1
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 1
1 0 2 2
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 1
1 0 2 1
```

```
1 0 1 0
1 1 0 1
0 1 1 0
0 1 2 2
1 0 2 2
```

```
1 0 1 0
1 1 0 0
0 1 1 1
0 1 2 1
1 1 1 2
```

```
1 0 1 0
1 1 0 1
0 1 1 2
0 1 2 1
1 1 1 2
```

```
1 0 1 0 1
1 1 0 0 2
0 1 1 1 0
0 1 2 1 2
```

```
1 0 1 0 1
1 1 0 0 2
0 1 1 1 2
0 1 2 1 0
```

```
1 0 1 0 1
1 1 0 0 1
0 1 1 1 0
0 1 2 1 2
```

```
1 0 1 0 1
1 1 0 0 1
0 1 1 1 0
0 1 2 1 1
```

```
1 0 1 0 1
1 1 0 1 1
0 1 1 0 2
0 1 2 2 2
```

```
1 0 1 0 1
1 1 0 1 0
0 1 1 2 2
0 1 2 0 0
```

```
1 0 1 0 1
1 1 0 1 1
0 1 1 1 0
0 1 2 0 1
```

```
1 0 1 0
1 1 0 0
0 1 1 1
1 1 2 1
1 2 1 1
```

```
1 0 1 1
1 1 0 1
0 1 1 2
1 1 2 2
1 2 1 2
```

```
1 0 1 1 1
1 1 0 1 2
0 1 1 2 1
1 1 2 2 0
```

```
1 0 1 0 0
1 1 0 0 1
0 1 1 1 0
1 1 1 2 2
```

## Appendix B

For interested readers we add a source listing of the Macek procedure we have used to generate our results in Theorem 5 and in Table 3.

```
# Before starting the procedure, create files "bw3-gfX-" containing
# the starting list of matroids to generate from, and "bw3-gfX+exc"
# containing extra excluded minors for bwidth3 over other fields.
# Then run a sequence of commands like these:
#  macek '&bw3excg gfX bw3 '
#  macek '&bw3excg gfX bw3 b'
#  ...
#  macek '&bw3excg gfX bw3 bbb...'
# Those will generate the excluded minors step-by-step,
# storing them to "bw3-gfX-b..b-exc".
@subd-param1  "gf3"
@subd-param2  "bw3"
@subd-param3  ""
@subd-param4  "b"
!pfield $param1
@sub-usefilen  ${param2}-${param1}
@sub-usefilenb ${usefilen}-${param3}
@sub-excextra  ${usefilen}+exc
@sub-treeall  ${usefilenb}-all
@sub-listin  ${usefilenb}
@sub-list3out  ${usefilenb}${param4}
@sub-list4out  ${usefilenb}${param4}-4
@sub-list4outn  ${list4out}n
@sub-exclist  ${listin}-exc
@sub-exclistout ${list3out}-exc
@sub-excluded   "((((S)(S)|"
@sub-excludedin  "((((1)("
@sub-excludedout "((((2)("
{
@name "bw3excg-w"
@comment "bw3excg (over $param1) working subframe:"
{
@name "exc-known"
@comment "known bw3 excl minors - extra, smaller, and new (generated)"
{
@name $excextra
!quiet
!iffile "$excextra"
!skip 1
!skip 4
!read $excextra
!filx-isompair ((s))
!pfield $param1
!represgen "((s))" allq >((0t))
}{
```

```
!quiet
!pfield $param1
!iffile "$exclist"
!mread $exclist >((0t))
}{ }
}{
@name extens1
@comment "all new ${param4}-extensions of input [${listin}]..."
}{
@name e-bwidth4
@comment "those generated with bwidth 4 get here:"
}{
@name e-bwidth4n
@comment "those new excl-minors with bwidth 4 get here:"
}{
@name e-bwidth3
@comment "those next with bwidth 3 get here:"
}}
@sub-input "(()(S))"
{
@inputpf $param1
<$listin
@comment "this is the starting set of matroids $listin:"
}
@sub-gener3  "(((4)("
@sub-generall  "(((1)("
@sub-gener4bw  "(((2)("
@sub-gener4n  "(((3)("
@extinherit ext-forbid
!extend b $input  >$generall(0t)|
!move ${generall}S|  >${gener3}(0t)|
!rex-bwidth3 ${gener3}S|
!move ^1  >${gener4bw}(0t)|
!writetreeto ${list3out} ${gener3}T|
!iflist 0 "<" ${gener4bw}S|
!writetreeto ${list4out} ${gener4bw}T|
!move ${gener4bw}S|  >$gener4n(0t)|
!filx-minor ${gener4n}s| $excluded
!iflist 0 "<" ${gener4n}S|
!writetreeto ${list4outn} ${gener4n}T|
!move ${excludedin}S| >$excludedout(0t)|
!move ${gener4n}S| >$excludedout(0t)|
!iflist 0 "<" ${excludedout}S|
!writetreeto ${exclistout} ${excludedout}T|
!writetreeto ${treeall} (T)
```