In order to obtain better bounds on the number of fully legal matchings, we increase the number of auxiliary quantities. Specifically, we determine an upper bound on the growth of the number of matchings without illegal components of at most four vertices (4-legal matchings). For every rooted tree T, we define the following quantities: e(T) is 1 if T is the empty tree, and 0 otherwise. Next, a_{ij}(T) is the number of 4-legal matchings where the root is not covered, the root component has i vertices and is adjacent to j matching edges. We only need this quantity for five different pairs (i,j): (1,0), (2,1), (3,2), (4,2), (4,3). All other 4-legal matchings that do not cover the root are counted by a(T). The reason only these five pairs play a role is that they are the only ones that can lead to an illegal component of at most four vertices. Likewise, b_{ij}(T) is the number of 4-legal matchings where the root is covered, the root component has i vertices and is adjacent to j matching edges. We only need this quantity for five different pairs (i,j): (1,1), (2,1), (2,2), (3,2), (3,3). All other 4-legal matchings that cover the root are counted by b(T). The following bilinear map computes the 13-dimensional vector of these quantities from the two vectors associated with the branches.

```
In[1]:= B[{ex_, a10x_, a21x_, a32x_, a42x_, a43x_, b11x_, b21x_, b22x_, b32x_, b33x_, ax_, bx_},
    {ey_, a10y_, a21y_, a32y_, a42y_, a43y_, b11y_, b21y_, b22y_, b32y_, b33y_, ay_,
     by_}] := {0, ex ey, b11x ey + ex b11y, b11x b11y + b22x ey + ex b22y, a21x b11y + b11x a21y +
     b11x b21y + b21x b11y + a10x b22y + b22x a10y + a32x ey + ex a32y + b32x ey + ex b32y,
    b11x b22y + b22x b11y + b33x ey + ex b33y, ex (a10y + ay) + (a10x + ax) ey,
    a10x (a10y + ay) + (a10x + ax) a10y, b11x (a10y + ay) + (a10x + ax) b11y,
    (a21x + b21x) (a10y + ay) + (a10x + ax) (a21y + b21y), b22x (a10y + ay) + (a10x + ax) b22y,
    (ex + a10x + a21x + a32x + a42x + a43x + b11x + b21x + b22x + b32x + b33x + ax + bx)
      (ey + a10y + a21y + a32y + a42y + a43y + b11y + b21y + b22y + b32y + b33y + ay + by) -
     (ex ey + b11x ey + ex b11y + b11x b11y + b22x ey + ex b22y + a21x b11y +
       b11x a21y + b11x b21y + b21x b11y + a10x b22y + b22x a10y + a32x ey +
       ex a32y + b32x ey + ex b32y + b11x b22y + b22x b11y + b33x ey + ex b33y),
    (ex + a10x + a21x + a32x + a42x + a43x + b11x + b21x + b22x + b32x + b33x + ax + bx) (a10y + ay) +
     (a10x + ax) (ey + a10y + a21y + a32y + a42y + a43y + b11y + b21y + b22y + b32y +
        b33y + ay + by) - (ex (a10y + ay) + (a10x + ax) ey + a10x (a10y + ay) +
       (a10x + ax) a10y + b11x (a10y + ay) + (a10x + ax) b11y + (a21x + b21x) (a10y + ay) +
       (a10x + ax) (a21y + b21y) + b22x (a10y + ay) + (a10x + ax) b22y)};
```

We initialize with the vector that is associated with the empty tree.

```
In[2]:= init = {1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
```

The following function determines by means of linear programming whether a vector v lies in the lower convex hull (the set of nonnegative vectors bounded above by the convex hull) determined by a set of vectors V.

```
In[3]:= InConvHull[v_, V_] :=
  Plus @@ (LinearProgramming[Table[1, {i, 1, Length[V]}], Transpose[V], v]) ≤ 1
```

The next function determines a minimal set of vectors for the lower convex hull.

```
In[4]:= ConvHull[V_] := Module[{L = V, n = Length[V]}, Do[
    If[InConvHull[L[[n + 1 - i]], Delete[L, n + 1 - i]], L = Delete[L, n + 1 - i]], {i, 1, n}];
   L]
```

Following Rosenfeld’s paper, we start with a single vector and keep updating the set of vectors until we reach a set X with the property that B(v,w) is in the lower convex hull of X for all v,w in X. We take alpha = 19/12, which will eventually result in a bound of the form O((19/12)^n).

In[5]:= `alpha = 19 / 12;`

In[6]:= `Clear[X]`

In[7]:= `X[0] = {init / alpha}`

Out[7]= $\left\{\left\{\frac{12}{19}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right\}\right\}$

In[8]:= `X[n_] := X[n] = ConvHull[Union[X[n - 1], Flatten[Outer[B, X[n - 1], X[n - 1], 1], 1]]]`

In[9]:= `Do[Print[Length[X[n]]], {n, 1, 7}]`

... LinearProgramming: No solution can be found that satisfies the constraints.

... Thread: Objects of unequal length in

$\{1\} + \left\{\{0\}, \left\{\frac{144}{361}\right\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}\right\} + \left\{\frac{12}{19}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right\}$ cannot be combined.

... LinearProgramming: No solution can be found that satisfies the constraints.

... Thread: Objects of unequal length in

$\{1\} + \left\{\left\{\frac{12}{19}\right\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}, \{0\}\right\} + \left\{0, \frac{144}{361}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right\}$ cannot be combined.

2

... LinearProgramming: No solution can be found that satisfies the constraints.

... General: Further output of LinearProgramming::lpsnf will be suppressed during this calculation.

... Thread: Objects of unequal length in

$\{1, 1, 1\} + \left\{\{0, 0, 0\}, \left\{0, 0, \frac{144}{361}\right\}, \{0, 0, 0\}, \{0, 0, 0\}, \{0, 0, 0\}, \{0, 0, 0\}, \left\{0, \frac{1728}{6859}, 0\right\}, \left\{\frac{41472}{130321}, 0, 0\right\}, \{0, 0, 0\}, \{0, 0, 0\}, \{0, 0, 0\}, \left\{\frac{20736}{130321}, \frac{1728}{6859}, 0\right\}, \{0, 0, 0\}\right\} + \left\{\frac{12}{19}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\right\}$ cannot be combined.

... General: Further output of Thread::tdlen will be suppressed during this calculation.

4

11

36

58

64

64

After seven steps, we reach a stable set of vectors.

In[10]:= `X[7] == X[6]`

Out[10]= True

These are the 64 vectors in question.

In[11]:= `X[7] // MatrixForm`

Out[11]//MatrixForm=

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	
0	0	0	0	0	0	0	$\frac{29\ldots}{47\ldots}$
0	0	0	0	0	0	0	$\frac{29\ldots}{47\ldots}$
0	0	0	0	0	0	0	$\frac{24\ldots}{24\ldots}$
0	0	0	0	0	0	0	$\frac{222902\ldots}{2213314\ldots}$
0	0	0	0	0	0	0	$\frac{4279728\ldots}{4205298\ldots}$
0	0	0	0	0	0	0	$\frac{8217078\ldots}{79900668\ldots}$
0	0	0	0	0	0	0	$\frac{4\ldots}{13\ldots}$
0	0	0	0	0	0	$\frac{248832}{2476099}$	
0	0	0	0	0	0	$\frac{1728}{6859}$	
0	0	0	0	0	$\frac{31060555494366117888}{5480386857784802185939}$	0	
0	0	0	0	0	$\frac{26748301344768}{2213314919066161}$	0	
0	0	0	0	0	$\frac{1486016741376}{116490258898219}$	0	
0	0	0	0	0	$\frac{215698302044209152}{15181127029874798299}$	0	
0	0	0	0	0	$\frac{1283918464548864}{42052983462257059}$	$\frac{6847565144260608}{42052983462257059}$	
0	0	0	0	0	$\frac{10319560704}{322687697779}$	0	
0	0	0	0	0	$\frac{10319560704}{322687697779}$	$\frac{30958682112}{322687697779}$	
0	0	0	0	$\frac{61917364224}{6131066257801}$	$\frac{61917364224}{6131066257801}$	0	
0	0	0	0	$\frac{26748301344768}{2213314919066161}$	$\frac{62412703137792}{2213314919066161}$	$\frac{356644017930240}{2213314919066161}$	
0	0	0	0	$\frac{2229025112064}{116490258898219}$	0	0	$\frac{10402\ldots}{116490\ldots}$
0	0	0	0	$\frac{123834728448}{6131066257801}$	0	0	
0	0	0	0	$\frac{17974858503684096}{799006685782884121}$	0	0	$\frac{3851755\ldots}{4205298\ldots}$
0	0	0	0	$\frac{429981696}{\ldots}$	$\frac{429981696}{\ldots}$	0	

$$
\left(\begin{array}{cccccccc}
0 & 0 & 0 & 0 & \frac{[illegible]}{16983563041} & \frac{[illegible]}{16983563041} & 0 & \\
0 & 0 & 0 & 0 & \frac{429981696}{16983563041} & \frac{429981696}{16983563041} & \frac{1719926784}{16983563041} & \\
0 & 0 & 0 & 0 & \frac{2972033482752}{116490258898219} & \frac{2972033482752}{116490258898219} & \frac{18575209267200}{116490258898219} & \\
0 & 0 & 0 & 0 & \frac{10319560704}{322687697779} & 0 & 0 & \\
0 & 0 & 0 & 0 & \frac{35831808}{893871739} & 0 & 0 & \frac{71\ldots}{893\ldots} \\
0 & 0 & 0 & 0 & \frac{309586821120}{6131066257801} & \frac{61917364224}{6131066257801} & \frac{928760463360}{6131066257801} & \\
0 & 0 & 0 & 0 & \frac{859963392}{16983563041} & 0 & 0 & \frac{1\,28\ldots}{16\,98\ldots} \\
0 & 0 & 0 & 0 & \frac{71663616}{893871739} & 0 & 0 & \\
0 & 0 & 0 & 0 & \frac{71663616}{893871739} & 0 & \frac{107495424}{893871739} & \\
0 & 0 & 0 & \frac{429981696}{16983563041} & 0 & 0 & 0 & \\
0 & 0 & 0 & \frac{185752092672}{6131066257801} & 0 & \frac{123834728448}{6131066257801} & \frac{866843099136}{6131066257801} & \\
0 & 0 & 0 & \frac{35831808}{893871739} & \frac{35831808}{893871739} & 0 & 0 & \\
0 & 0 & 0 & \frac{2985984}{47045881} & 0 & 0 & 0 & \\
0 & 0 & 0 & \frac{2985984}{47045881} & 0 & 0 & \frac{5971968}{47045881} & \\
0 & 0 & 0 & \frac{71663616}{893871739} & \frac{35831808}{893871739} & 0 & \frac{107495424}{893871739} & \\
0 & 0 & \frac{1289945088}{16983563041} & 0 & \frac{859963392}{16983563041} & 0 & \frac{2149908480}{16983563041} & \\
0 & 0 & \frac{71663616}{893871739} & 0 & \frac{35831808}{893871739} & 0 & \frac{143327232}{893871739} & \\
0 & 0 & \frac{20736}{130321} & 0 & 0 & 0 & \frac{20736}{130321} & \\
0 & \frac{144}{361} & 0 & 0 & 0 & 0 & 0 & \\
\frac{12}{19} & 0 & 0 & 0 & 0 & 0 & 0 &
\end{array}\right.
$$

We can check directly that the set satisfies the required condition.

In[12]:=
```
Z = X[7];
```

In[13]:=
```
Y = Union[Z, Flatten[Outer[B, Z, Z, 1], 1]];
```

Indeed, all vectors of the form B(v,w) lie in the lower convex hull. Now the same proof as for 2-legal matchings shows that the number of 4-legal matchings is O((19/12)^n).

In[14]:=
```
Union[Table[InConvHull[Y[[j]], Z], {j, 1, Length[Y]}]]
```

Out[14]= {True}