Examples

Richard Adhika

In this file, we run through some examples for ratio 1, 2, 3, and 4 (the code can be found in the following files: 'ratio1_and_ratio.ipynb', 'ratio3.ipynb', and 'ratio4.ipynb'). For outputs of ratio 2, the array represents increasing $kbig$ as specified in each example. For outputs of ratio 3 and ratio 4, the array represents increasing $e = 0, 1, \ldots$. Lastly, for ratio 4, we use $kbig$ to be the largest of the ones used in ratio 2 in the same example. The following runtime may be inaccurate as (for some unknown reason) running Julia in WSL (Windows Subsystem for Linux) can cause inconsistent runtime (as much as a factor of 2!). The examples are as follows,

1. $p = 2, n = 9, M' = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}$.

   For ratio 2 with $kbig = 10, \ldots, 17$, ratio 3 with $e = 0, \ldots, 5$, $k = n$, and ratio 4 with $kbig = 17$, we get

```
ratio 1: 2//1
  0.000409 seconds (1.69 k allocations: 32.148 KiB)
ratio 2 (kbig = 10, ..., 17): 3//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
  0.758604 seconds (11.63 M allocations: 189.849 MiB, 38.39% gc time)
ratio 3 (e = 0, ..., 5): 1//2, 3//2, 2//1, 2//1, 2//1, 2//1
  4.989241 seconds (22.33 M allocations: 1.742 GiB, 30.17% gc time)
ratio 4 (e = 0, 1, 2): 47971//196608, 244579//196608, 3//2
ratio 4 (e = 3, 4, 5): 3//2, 3//2, 3//2
  4.141612 seconds (26.21 M allocations: 1.391 GiB, 42.76% gc time)
```

   For ratio 3 with $k = 1, \ldots, 8$, we get

```
ratio 3 (e = 0,...,5):
k = 1 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 2 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 3 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 4 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 5 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 6 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 7 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 8 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
 16.417401 seconds (31.48 M allocations: 2.235 GiB, 36.59% gc time)
```

2. $p = 2, n = 11, M' = \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix}$.

   For ratio 2 with $kbig = 12, \ldots, 17$, ratio 3 with $e = 0, \ldots, 6$, $k = n$, and ratio 4 with $kbig = 17$, we get

```
ratio 1: 2//1
  0.178761 seconds (59.25 k allocations: 2.954 MiB, 99.41% compilation time)
ratio 2 (kbig = 12, ..., 17): 3//2, 11//8, 11//8, 11//8, 11//8, 11//8
  1.984675 seconds (14.25 M allocations: 373.781 MiB, 31.78% gc time, 12.51% compilation time)
ratio 3 (e = 0, ..., 6): 1//8, 3//8, 3//2, 7//4, 2//1, 2//1, 2//1
 89.033111 seconds (287.50 M allocations: 26.356 GiB, 28.62% gc time, 1.81% compilation time)
ratio 4 (e = 0, 1, 2, 3): 96313//3145728, 487709//3145728, 3732539//3145728, 4127575//3145728
ratio 4 (e = 4, 5, 6): 11//8, 11//8, 11//8
 54.781797 seconds (295.60 M allocations: 19.374 GiB, 33.74% gc time, 2.34% compilation time)
```

For ratio 3 with $k = 1, \ldots, 11$, we get

```
ratio 3 (e = 0,...,6):
k = 1 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 2 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 3 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 4 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 5 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 6 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 7 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 8 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 9 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 10 → 1//4, 3//4, 2//1, 2//1, 2//1, 2//1, 2//1
k = 11 → 1//8, 3//8, 3//2, 7//4, 2//1, 2//1, 2//1
110.031992 seconds (382.70 M allocations: 32.855 GiB, 30.48% gc time, 0.43% compilation time)
```

3. $p = 2, n = 11, M' = \begin{pmatrix} 1 & 0 \\ 0 & 9 \end{pmatrix}$.

For ratio 2 with $kbig = 12, \ldots, 17$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 17$, we get

```
ratio 1: 2//1
  0.000783 seconds (6.15 k allocations: 96.453 KiB)
ratio 2 (kbig = 12, ..., 17): 3//2, 11//8, 43//32, 43//32, 43//32, 43//32
  1.384114 seconds (14.20 M allocations: 339.110 MiB, 41.09% gc time)
ratio 3 (e = 0, ..., 7): 1//32, 3//32, 3//8, 23//16, 7//4, 15//8, 2//1, 2//1
 67.260180 seconds (249.29 M allocations: 25.314 GiB, 28.51% gc time, 0.17% compilation time)
ratio 4 (e = 0, 1, 2, 3): 4209//1048576, 20785//1048576, 468265//3145728, 457883//393216
ratio 4 (e = 4, 5, 6, 7): 2040211//1572864, 4178231//3145728, 43//32, 43//32
 50.211738 seconds (285.70 M allocations: 18.873 GiB, 33.83% gc time, 0.36% compilation time)
```

For ratio 3 with $k = 1, \ldots, 11$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//1, 2//1, 2//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 2 → 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 3 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 4 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 5 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 6 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 7 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 8 → 1//4, 1//2, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 9 → 1//8, 3//8, 1//1, 2//1, 2//1, 2//1, 2//1, 2//1
k = 10 → 1//16, 3//16, 3//4, 15//8, 2//1, 2//1, 2//1, 2//1
k = 11 → 1//32, 3//32, 3//8, 23//16, 7//4, 15//8, 2//1, 2//1
 93.045442 seconds (319.96 M allocations: 31.351 GiB, 29.75% gc time)
```

4. $p = 3, n = 6, M' = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}$.

   For ratio 2 with $kbig = 7, \ldots, 12$, ratio 3 with $e = 0, \ldots, 6$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 3//2
  0.014180 seconds (4.37 k allocations: 153.984 KiB, 98.24% compilation time)
ratio 2 (kbig = 7, ..., 12): 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
  0.882785 seconds (21.15 M allocations: 348.741 MiB, 6.87% compilation time)
ratio 3 (e = 0, ..., 6): 1//6, 7//6, 3//2, 3//2, 3//2, 3//2, 3//2
  9.982162 seconds (32.78 M allocations: 2.992 GiB, 38.70% gc time, 1.11% compilation time)
ratio 4 (e = 0, 1, 2, 3): 8401//157464, 165865//157464, 7//6, 7//6
ratio 4 (e = 4, 5, 6): 7//6, 7//6, 7//6
  7.866748 seconds (49.57 M allocations: 2.602 GiB, 39.25% gc time, 2.34% compilation time)
```

   For ratio 3 with $k = 1, \ldots, 6$, we get

```
ratio 3 (e = 0,...,6):
k = 1 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 2 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 3 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 4 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 5 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 6 → 1//6, 7//6, 3//2, 3//2, 3//2, 3//2, 3//2
 20.793037 seconds (37.54 M allocations: 3.303 GiB, 39.90% gc time)
```

5. $p = 3, n = 7, M' = \begin{pmatrix} 1 & 0 \\ 0 & 10 \end{pmatrix}$.

   For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 3//2
  0.009965 seconds (8.70 k allocations: 221.672 KiB, 97.04% compilation time)
ratio 2 (kbig = 8, ..., 12): 7//6, 61//54, 61//54, 61//54, 61//54
  1.927119 seconds (24.44 M allocations: 447.070 MiB, 37.40% gc time, 3.07% compilation time)
ratio 3 (e = 0, ..., 7): 1//54, 7//54, 7//6, 25//18, 3//2, 3//2, 3//2, 3//2
 64.736099 seconds (239.46 M allocations: 25.175 GiB, 27.38% gc time, 0.17% compilation time)
ratio 4 (e = 0, 1, 2, 3): 7903//4251528, 161213//4251528, 1476607//1417176, 4748903//4251528
ratio 4 (e = 4, 5, 6, 7): 61//54, 61//54, 61//54, 61//54
 55.421852 seconds (313.24 M allocations: 20.816 GiB, 33.45% gc time, 0.32% compilation time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 2 → 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 3 → 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 4 → 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 5 → 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 6 → 1//18, 7//18, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 7 → 1//54, 7//54, 7//6, 25//18, 3//2, 3//2, 3//2, 3//2
140.136758 seconds (265.94 M allocations: 27.556 GiB, 26.11% gc time)
```

6. $p = 3, n = 7, M' = \begin{pmatrix} 1 & 0 \\ 0 & 28 \end{pmatrix}$.

For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 3//2
  0.079259 seconds (59.53 k allocations: 2.958 MiB, 99.04% compilation time)
ratio 2 (kbig = 8, ..., 12): 7//6, 61//54, 547//486, 547//486, 547//486
  2.538242 seconds (24.67 M allocations: 556.715 MiB, 34.14% gc time, 5.20% compilation time)
ratio 3 (e = 0, ..., 7): 1//486, 7//486, 7//54, 187//162, 25//18, 79//54, 3//2, 3//2
 71.670046 seconds (221.90 M allocations: 24.729 GiB, 25.62% gc time, 1.57% compilation time)
ratio 4 (e = 0, 1, 2, 3): 233//2125764, 37//26244, 54007//1417176, 4424993//4251528
ratio 4 (e = 4, 5, 6, 7): 2371843//2125764, 2389601//2125764, 547//486, 547//486
 59.188280 seconds (312.02 M allocations: 20.774 GiB, 33.08% gc time, 1.64% compilation time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 2 → 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 3 → 1//18, 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 4 → 1//18, 1//6, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 5 → 1//54, 7//54, 1//2, 3//2, 3//2, 3//2, 3//2, 3//2
k = 6 → 1//162, 7//162, 7//18, 79//54, 3//2, 3//2, 3//2, 3//2
k = 7 → 1//486, 7//486, 7//54, 187//162, 25//18, 79//54, 3//2, 3//2
127.481749 seconds (241.30 M allocations: 26.932 GiB, 25.31% gc time)
```

7. $p = 3, n = 7, M' = \begin{pmatrix} 1 & 0 \\ 0 & 82 \end{pmatrix}$.

For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 107//72
  0.000530 seconds (6.64 k allocations: 109.594 KiB)
ratio 2 (kbig = 8, ..., 12): 755//648, 61//54, 547//486, 4921//4374, 4921//4374
  2.451098 seconds (24.19 M allocations: 548.482 MiB, 41.05% gc time)
ratio 3 (e = 0, ..., 7): 1//4374, 7//4374, 7//486, 187//1458, 187//162, 673//486, 947//648, 947//648
 54.935133 seconds (213.17 M allocations: 24.316 GiB, 26.49% gc time, 0.23% compilation time)
ratio 4 (e = 0, 1, 2, 3): 53//2125764, 433//4251528, 86//59049, 20027//531441
ratio 4 (e = 4, 5, 6, 7): 2211629//2125764, 1185821//1062882, 4778741//4251528, 398549//354294
 55.422650 seconds (310.77 M allocations: 20.716 GiB, 33.25% gc time, 0.34% compilation time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 35//72, 107//72, 107//72, 107//72, 107//72, 107//72, 107//72, 107//72
k = 2 → 11//72, 35//72, 107//72, 107//72, 107//72, 107//72, 107//72, 107//72
k = 3 → 1//24, 11//72, 35//72, 107//72, 107//72, 107//72, 107//72, 107//72
k = 4 → 1//216, 1//24, 11//72, 35//72, 107//72, 107//72, 107//72, 107//72
k = 5 → 1//486, 7//486, 83//648, 35//72, 107//72, 107//72, 107//72, 107//72
k = 6 → 1//1458, 7//1458, 7//162, 187//486, 947//648, 107//72, 107//72, 107//72
k = 7 → 1//4374, 7//4374, 7//486, 187//1458, 187//162, 673//486, 947//648, 947//648
120.290833 seconds (231.96 M allocations: 26.541 GiB, 24.07% gc time)
```

In this example, the reason why ratio 3 isn't eventually equal to ratio 1 is because the minimum 3-valuation of the conjugacy matrix of some matrices is greater than or equal to 7. Therefore, since $n = 7$, those determinant is treated as zero and doesn't contribute to the count. Higher n is needer to properly evaluate ratio 3 in this example.

8. $p = 3, n = 7, M' = \begin{pmatrix} 1 & 6 \\ 3 & 1 \end{pmatrix}$.

For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 5//4
  0.090920 seconds (59.70 k allocations: 2.961 MiB, 99.01% compilation time)
ratio 2 (kbig = 8, ..., 12): 13//12, 13//12, 13//12, 13//12, 13//12
  2.088093 seconds (21.39 M allocations: 339.791 MiB, 33.72% gc time, 7.43% compilation time)
ratio 3 (e = 0, ..., 7): 1//12, 13//12, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
102.521372 seconds (425.35 M allocations: 23.613 GiB, 31.37% gc time, 1.25% compilation time)
ratio 4 (e = 0, 1, 2, 3): 118087//4251528, 4369615//4251528, 13//12, 13//12
ratio 4 (e = 4, 5, 6, 7): 13//12, 13//12, 13//12, 13//12
 62.158459 seconds (340.48 M allocations: 20.614 GiB, 31.55% gc time, 1.68% compilation time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 2 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 3 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 4 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 5 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 6 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 7 → 1//12, 13//12, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
197.453824 seconds (470.38 M allocations: 26.158 GiB, 29.32% gc time)
```

9. $p = 3, n = 7, M' = \begin{pmatrix} 1 & 18 \\ 9 & 1 \end{pmatrix}$.

For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 17//12
  0.000766 seconds (6.62 k allocations: 104.000 KiB)
ratio 2 (kbig = 8, ..., 12): 41//36, 121//108, 121//108, 121//108, 121//108
  2.532818 seconds (22.98 M allocations: 369.506 MiB, 47.74% gc time)
ratio 3 (e = 0, ..., 7): 1//108, 13//108, 41//36, 49//36, 17//12, 17//12, 17//12, 17//12
 82.712178 seconds (296.17 M allocations: 24.956 GiB, 28.62% gc time)
ratio 4 (e = 0, 1, 2, 3): 4099//4251528, 53195//1417176, 4419781//4251528, 4736687//4251528
ratio 4 (e = 4, 5, 6, 7): 121//108, 121//108, 121//108, 121//108
 68.953079 seconds (355.36 M allocations: 21.311 GiB, 32.28% gc time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 5//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 2 → 1//12, 5//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 3 → 1//12, 5//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 4 → 1//12, 5//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 5 → 1//12, 5//12, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 6 → 1//36, 13//36, 17//12, 17//12, 17//12, 17//12, 17//12, 17//12
k = 7 → 1//108, 13//108, 41//36, 49//36, 17//12, 17//12, 17//12, 17//12
156.057766 seconds (325.22 M allocations: 27.336 GiB, 25.66% gc time)
```

10. $p = 3, n = 7, M' = \begin{pmatrix} 2 & -2 \\ 1 & 0 \end{pmatrix}$.

For ratio 2 with $kbig = 8, \ldots, 12$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 12$, we get

```
ratio 1: 3//4
  0.000629 seconds (6.72 k allocations: 111.141 KiB)
ratio 2 (kbig = 8, ..., 12): 3//4, 3//4, 3//4, 3//4, 3//4
  0.564643 seconds (16.67 M allocations: 254.400 MiB)
ratio 3 (e = 0, ..., 7): 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
 89.072864 seconds (503.93 M allocations: 19.064 GiB, 40.61% gc time)
ratio 4 (e = 0, 1, 2, 3): 3//4, 3//4, 3//4, 3//4
ratio 4 (e = 4, 5, 6, 7): 3//4, 3//4, 3//4, 3//4
 44.369667 seconds (202.62 M allocations: 13.908 GiB, 34.74% gc time)
```

For ratio 3 with $k = 1, \ldots, 7$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 2 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 3 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 4 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 5 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 6 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
k = 7 → 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4, 3//4
182.135902 seconds (564.57 M allocations: 21.407 GiB, 36.42% gc time)
```

11. $p = 5, n = 4, M' = \begin{pmatrix} 1 & 0 \\ 0 & 6 \end{pmatrix}$.

For ratio 2 with $kbig = 5, \ldots, 9$, ratio 3 with $e = 0, \ldots, 5$, $k = n$, and ratio 4 with $kbig = 9$, we get

```
ratio 1: 5//4
  0.000217 seconds (1.90 k allocations: 30.031 KiB)
ratio 2 (kbig = 5, ..., 9): 21//20, 21//20, 21//20, 21//20, 21//20
  3.611637 seconds (39.42 M allocations: 614.310 MiB, 45.19% gc time)
ratio 3 (e = 0, ..., 5): 1//20, 21//20, 5//4, 5//4, 5//4, 5//4
  4.378132 seconds (18.64 M allocations: 1.812 GiB, 30.78% gc time)
ratio 4 (e = 0, 1, 2, 3): 1009//125000, 126009//125000, 21//20
ratio 4 (e = 3, 4, 5, 7): 21//20, 21//20, 21//20
  7.589193 seconds (56.34 M allocations: 2.199 GiB, 41.97% gc time)
```

For ratio 3 with $k = 1, \ldots, 4$, we get

```
ratio 3 (e = 0,...,5):
k = 1 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 2 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 3 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 4 → 1//20, 21//20, 5//4, 5//4, 5//4, 5//4
 10.359664 seconds (19.50 M allocations: 1.881 GiB, 35.90% gc time)
```

12. $p = 5, n = 5, M' = \begin{pmatrix} 1 & 0 \\ 0 & 26 \end{pmatrix}$.

For ratio 2 with $kbig = 6, \ldots, 9$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 9$, we get

```
ratio 1: 5//4
  0.000844 seconds (9.39 k allocations: 152.484 KiB)
ratio 2 (kbig = 6, ..., 9): 21//20, 521//500, 521//500, 521//500
  4.401260 seconds (48.18 M allocations: 899.485 MiB, 43.90% gc time)
ratio 3 (e = 0, ..., 7): 1//500, 21//500, 21//20, 121//100, 5//4, 5//4, 5//4, 5//4
112.057273 seconds (401.68 M allocations: 45.590 GiB, 25.23% gc time)
ratio 4 (e = 0, 1, 2, 3): 883//9375000, 73861//9375000, 9451609//9375000, 9753631//9375000
ratio 4 (e = 4, 5, 6, 7): 521//500, 521//500, 521//500, 521//500
121.516638 seconds (610.17 M allocations: 41.476 GiB, 36.23% gc time)
```

For ratio 3 with $k = 1, \ldots, 5$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 2 → 1//20, 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 3 → 1//20, 1//4, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 4 → 1//100, 21//100, 5//4, 5//4, 5//4, 5//4, 5//4, 5//4
k = 5 → 1//500, 21//500, 21//20, 121//100, 5//4, 5//4, 5//4, 5//4
224.429634 seconds (415.53 M allocations: 47.142 GiB, 22.49% gc time)
```

13. $p = 5, n = 5, M' = \begin{pmatrix} 1 & 0 \\ 0 & 126 \end{pmatrix}$.

For ratio 2 with $kbig = 6, \ldots, 9$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 9$, we get

```
ratio 1: 149//120
  0.001498 seconds (9.46 k allocations: 153.938 KiB)
ratio 2 (kbig = 6, ..., 9): 3149//3000, 521//500, 13021//12500, 13021//12500
  4.147181 seconds (47.74 M allocations: 893.386 MiB, 41.44% gc time)
ratio 3 (e = 0, ..., 7): 1//12500, 21//12500, 21//500, 2621//2500, 3629//3000, 3629//3000, 3629//3000, 3629//3000
117.294749 seconds (387.12 M allocations: 44.998 GiB, 26.03% gc time)
ratio 4 (e = 0, 1, 2, 3): 49//4687500, 839//9375000, 9196//1171875, 4725353//4687500
ratio 4 (e = 4, 5, 6, 7): 2438261//2343750, 651011//625000, 13021//12500, 13021//12500
126.969844 seconds (609.68 M allocations: 41.459 GiB, 32.95% gc time)
```

For ratio 3 with $k = 1, \ldots, 5$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 29//120, 149//120, 149//120, 149//120, 149//120, 149//120, 149//120, 149//120
k = 2 → 1//24, 29//120, 149//120, 149//120, 149//120, 149//120, 149//120, 149//120
k = 3 → 1//600, 1//24, 29//120, 149//120, 149//120, 149//120, 149//120, 149//120
k = 4 → 1//2500, 21//2500, 629//3000, 149//120, 149//120, 149//120, 149//120, 149//120
k = 5 → 1//12500, 21//12500, 21//500, 2621//2500, 3629//3000, 3629//3000, 3629//3000, 3629//3000
226.686866 seconds (399.42 M allocations: 46.519 GiB, 27.74% gc time)
```

Note that this example has the same issue as example 7.

14. $p = 5, n = 5, M' = \begin{pmatrix} 1 & 10 \\ 5 & 1 \end{pmatrix}$.

For ratio 2 with $kbig = 6, \ldots, 8$, ratio 3 with $e = 0, \ldots, 7$, $k = n$, and ratio 4 with $kbig = 8$, we get

```
ratio 1: 7//6
  0.117796 seconds (62.51 k allocations: 3.003 MiB, 99.01% compilation time)
ratio 2 (kbig = 6, ..., 8): 31//30, 31//30, 31//30
  1.110414 seconds (9.34 M allocations: 160.030 MiB, 27.84% gc time, 20.31% compilation time)
ratio 3 (e = 0, ..., 7): 1//30, 31//30, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
167.947148 seconds (653.20 M allocations: 44.262 GiB, 31.31% gc time, 0.87% compilation time)
```

```
ratio 4 (e = 0, 1, 2, 3): 61609//9375000, 9436609//9375000, 31//30, 31//30
ratio 4 (e = 4, 5, 6, 7): 31//30, 31//30, 31//30, 31//30
127.299724 seconds (653.73 M allocations: 41.898 GiB, 34.97% gc time, 2.56% compilation time)
```

For ratio 3 with $k = 1, \ldots, 5$, we get

```
ratio 3 (e = 0,...,7):
k = 1 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 2 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 3 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 4 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 5 → 1//30, 31//30, 7//6, 7//6, 7//6, 7//6, 7//6, 7//6
352.014799 seconds (675.44 M allocations: 45.900 GiB, 27.03% gc time)
```

15. $p = 7, n = 4, M' = \begin{pmatrix} 1 & 0 \\ 0 & 8 \end{pmatrix}$.

For ratio 2 with $kbig = 5, \ldots, 7$, ratio 3 with $e = 0, \ldots, 5$, $k = n$, and ratio 4 with $kbig = 7$, we get

```
ratio 1: 7//6
  0.076057 seconds (60.30 k allocations: 2.970 MiB, 98.96% compilation time)
ratio 2 (kbig = 5, ..., 7): 43//42, 43//42, 43//42
  1.126612 seconds (14.95 M allocations: 274.204 MiB, 24.57% gc time, 13.20% compilation time)
ratio 3 (e = 0, ..., 5): 1//42, 43//42, 7//6, 7//6, 7//6, 7//6
 65.217971 seconds (248.43 M allocations: 25.752 GiB, 28.22% gc time, 2.58% compilation time)
```

```
ratio 4 (e = 0, 1, 2, 3): 1507//470596, 472103//470596, 43//42
ratio 4 (e = 3, 4, 5, 6): 43//42, 43//42, 43//42
 63.291128 seconds (345.20 M allocations: 25.480 GiB, 33.13% gc time, 1.58% compilation time)
```

For ratio 3 with $k = 1, \ldots, 5$, we get

```
ratio 3 (e = 0,...,5):
k = 1 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 2 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 3 → 1//6, 7//6, 7//6, 7//6, 7//6, 7//6
k = 4 → 1//42, 43//42, 7//6, 7//6, 7//6, 7//6
144.528261 seconds (252.43 M allocations: 26.138 GiB, 29.03% gc time)
```