Meccano octagons

https://github.com/heptagons/meccano/octa

1 Meccano regular octagons

Meccano regular octagons are build with eight equals rods to form the perimeter and several extra rods as internal diagonals to make rigid the octagon's internal angles equal to 135°. Figure 1 show the internal diagonals construction. The idea is to form two adjacent triangles with two angles adding to 145°. Consider triangle ABC with $\angle BCA = 45°$ and triangle ACD_n with $\angle ACD_n = 90°$, so $\angle BCE = 135°$. Lets define:

$$a = \overline{AD_n}$$
$$b = \overline{CD_n}$$
$$c = \overline{AB} = \overline{BC} = \overline{CE}$$

The internal diagonal we are looking for is the integer a (shown in red) and the two adjacent octagon sides are $c = \overline{BC} = \overline{CE}$ (shown in green). The common hyphotenuse shown in the figure as a dashed line value is $\overline{AC} = \sqrt{2}c$ so:

$$(\sqrt{2}c)^2 = a^2 - b^2$$
$$2c^2 = a^2 - b^2$$
$$c = \sqrt{\frac{a^2 - b^2}{2}}$$

We need a program to test pairs $a > b$ which makes a c an integer too.

1.1 Octagons diagonals program

Next golang listing program finds valid octagons diagonals a and sides $\max(b, c)$. First we iterate diagonals from 1 to a given maximum (line 2). Then we iterate over integer b from 1 to a (line 3). We calculate

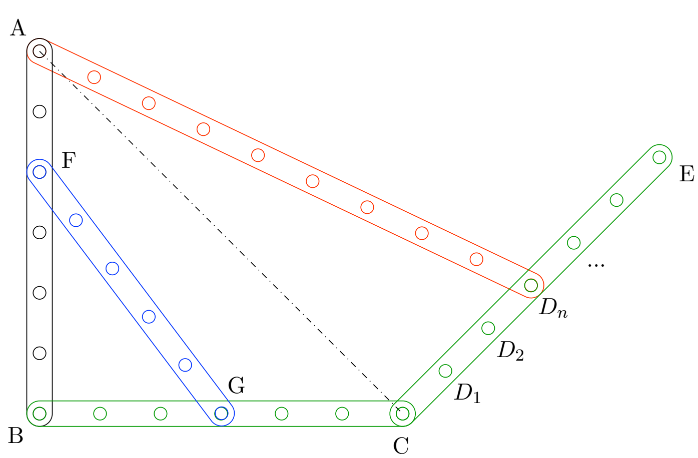

Figure 1: plan

$2c^2$ and check if its even (lines 4, 5) and the we check if c is an integer (line 8). To prevent repetitions by scaling we check the greatest common divisor to be 1 (line 9) and print a valid result where the octagon size is the maximum value of b or c. (line 11).

```
1  func octagons_2(max int) {
2      for a := 1; a < max; a++ {
3          for b := 1; b < a; b++ {
4              cc := a*a - b*b
5              if cc % 2 == 0 {
6                  f := math.Sqrt(float64(cc/2))
7                  c := int(f)
8                  if f == float64(c) {
9                      if gcd(c, gcd(b, a)) == 1 {
10                         s := int(math.Max(float64(b), f))
11                         fmt.Printf("a=%2d b=%2d c=%2d s=%2d\n", a, b, c, s)
12                     }
13                 }
14             }
15         }
16     }
17 }
18
19 func gcd(a, b int) int { // greatest common divisor
20     if b == 0 {
21         return a
22     }
23     return gcd(b, a % b)
24 }
```

1.2 Octagons diagonals results

Program's first results are shown in the next listing. Important integers are a the diagonal and s the size.

```
a= 3 b= 1 c= 2 s= 2
a= 9 b= 7 c= 4 s= 7
a=11 b= 7 c= 6 s= 7
a=17 b= 1 c=12 s=12
a=19 b=17 c= 6 s=17
a=27 b=23 c=10 s=23
a=33 b=17 c=20 s=20
a=33 b=31 c= 8 s=31
a=41 b=23 c=24 s=24
a=43 b= 7 c=30 s=30
a=51 b=47 c=14 s=47
a=51 b=49 c=10 s=49
a=57 b= 7 c=40 s=40
a=57 b=41 c=28 s=41
a=59 b=41 c=30 s=41
```

1.3 Octagons with diagonal=6

We can't use the first result $a = 3$, $b = 1$, $c = 2$ and $s = 2$ since s is to small. To make rigid the angle of 90° of triangle ABC in figure 1 we need a rod of size 5 at least, as shown the points F and G of the figure. Scaling by 2 the first result we have a diagonal $a = 6$ and size=4 as a base to build the smaller octagons.

Figure 2 is the smallest octagon; we need to scale the bars in order the bolts don't collapse with others, the complexity of bars can be simplified symmetrically as is show in figure 6. In figure 3 we increase the

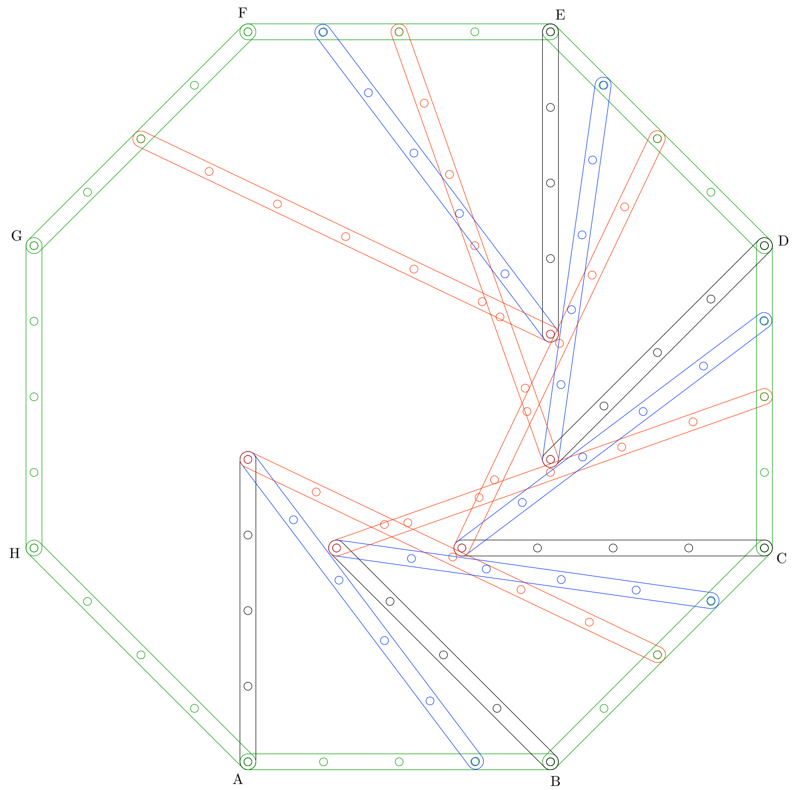

Figure 2: The smallest octagon with diagonals 6 and sides 4.

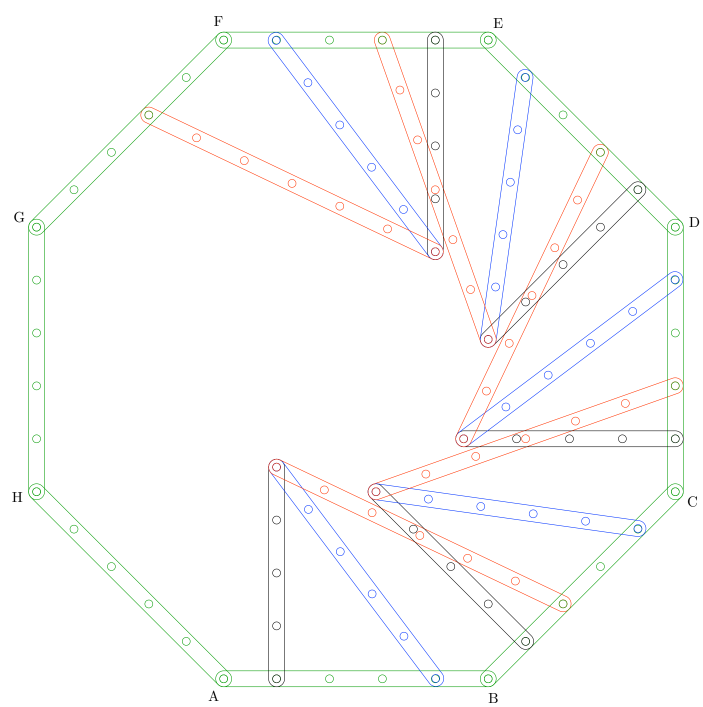

Figure 3: Octagon with diagonals 6 and sides 5.

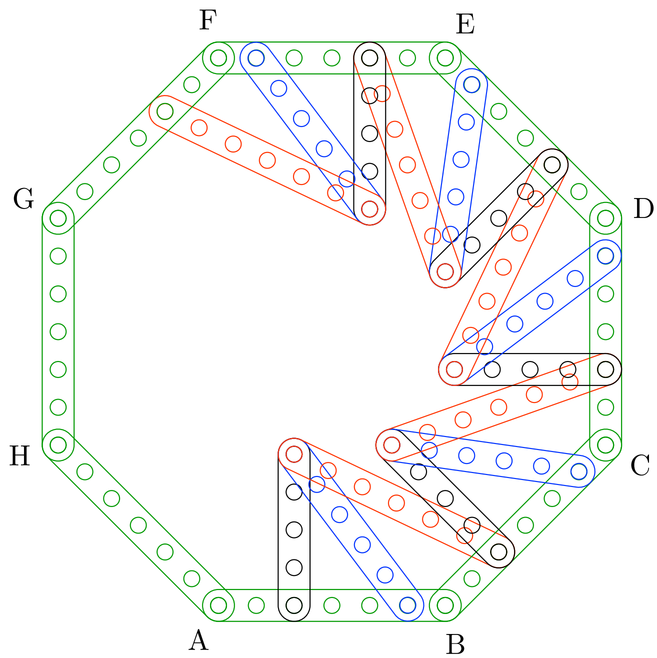

Figure 4: Octagon with diagonals 6 and sides 6.

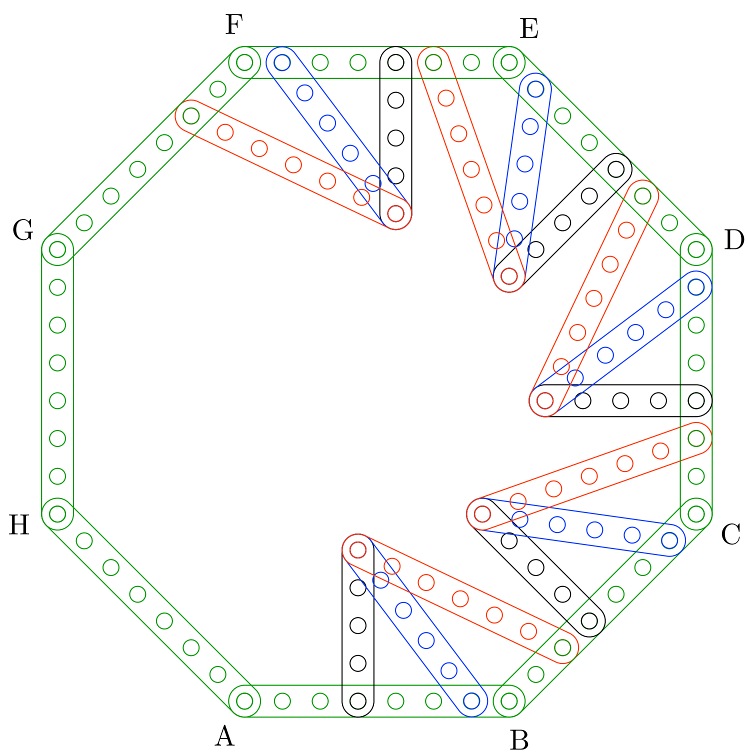

Figure 5: Octagon with diagonals 6 and sides 7.

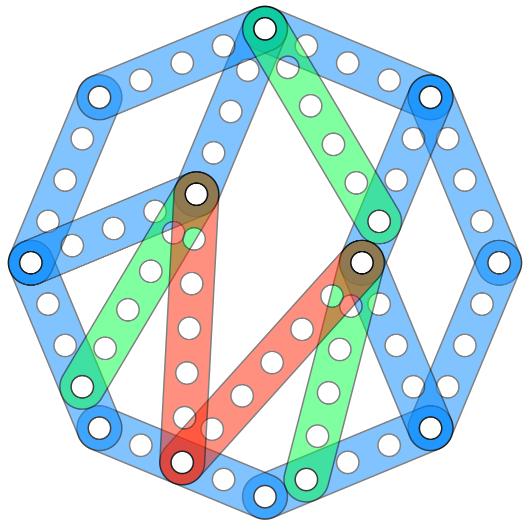

Figure 6: The smallest octagon with sides 4 and with fewest diagonals.

side from 4 to 5 but keeping the same diagonal of 6. In figure 4 we increase the side to 6 and in figure 5 the side is increased to 7.

1.4 Octagons with diagonal=9

Using the second result $a = 9$, $b = 7$, $c = 4$ and $s = 7$ we form a second group of octagons. Figure 7 shows the smallest octagon with diagonals 9 and sides 7. Figures 8, 9, 10 and 11 show octagons with diagonals 9 with sides of 8, 9, 10 and 11.

1.5 Octagon with two diagonals

By comparing figures 5 and 10 both with sides=7 we can make use of two diagonals at the same time and omit the rod FG of figure 1 used until now to make the 90° angle. Figure 12 show a octagon angle with two diagonals. In other words, for two results or their scaling, when both have the same c, we can use the two diagonals an omit the 5 rod.

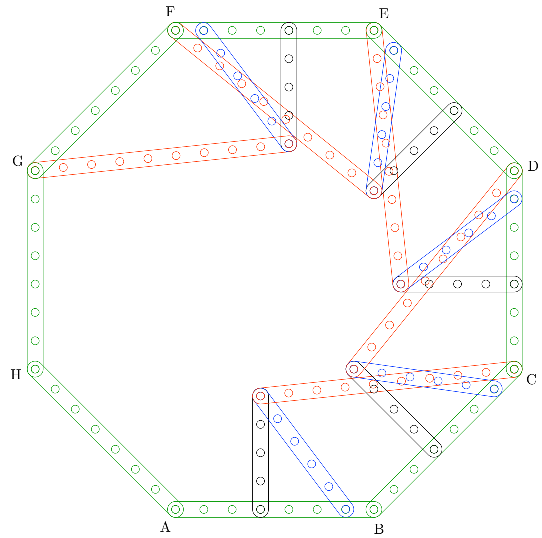

Figure 7: Octagon with diagonal 9 and side 7.

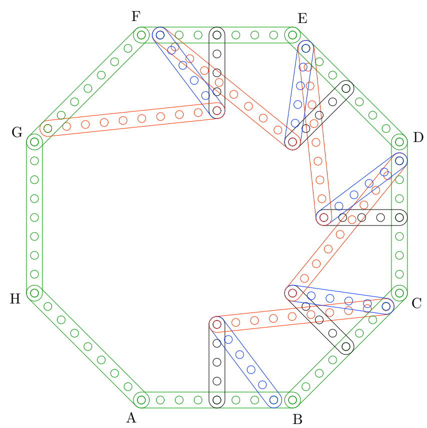

Figure 8: Octagon with diagonal 9 and side 8.

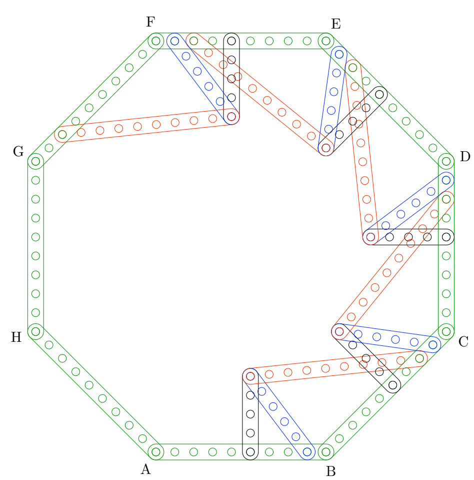

Figure 9: Octagon with diagonal 9 and side 9.

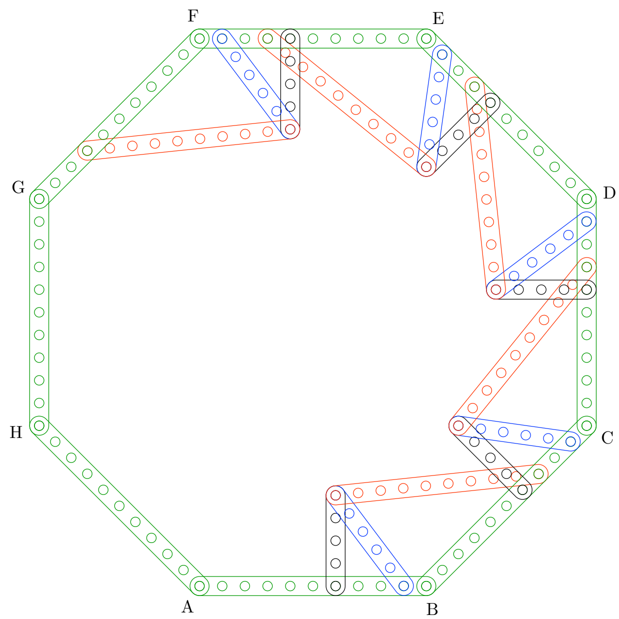

Figure 10: Octagon with diagonal 9 and side 10.

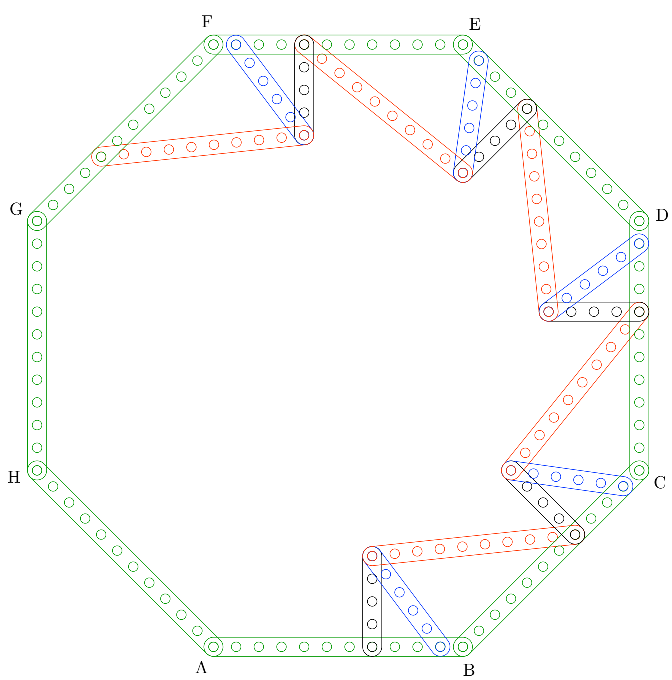

Figure 11: Octagon with diagonal 9 and side 11.

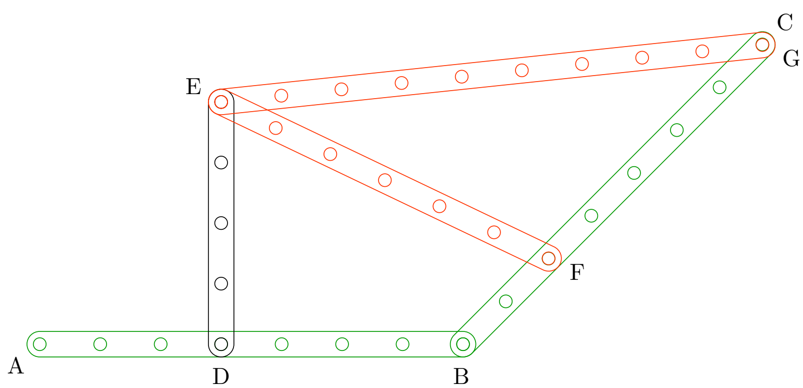

Figure 12: Octagon angle ABC fixed with two diagonals. The union of rods $\overline{BG}$, $\overline{EF}$ and $\overline{EG}$ is rigid. Adding two rods $\overline{AB}$ and $\overline{DE}$ remains rigid.