3x3 MAGIC SQUARES: CUBICAL AND AFFINE

Constituents of the counting functions

and initial values of the functions,

by geometry and generating functions

The number of terms to evaluate in the series expansions, to find the actual numbers of squares and symmetry types, M_c(t), m_c(t), M_a(t), m_a(t).

Small values, like 1000, are computed quickly. Large values, like 10000 (cubic) or 30000 (affine) take noticeably longer.

```
> enddegree:=100;
```

$$enddegree := 100$$

Magic squares: cubical count

The dimension and period for cubical magic, and the standard denominator:

```
> d:=3:
  p:=12:
  cdenom:=(1-x^p)^(d+1):
```

See Section 2.1 of "Six Little Squares". We get LattE to spit out two generating functions--one for the polytope P_c:

```
> ehrpc:= 1 / ((1-t) * (1-t) * (1-t^2) * (1-t^4)):
```

... and one for h intersect P_c:

```
> ehrhc:= 1 / ((1-t) * (1-t) * (1-t^6)):
```

LattE gives the closed generating functions, which we need to convert into their open counterparts:

```
> openehrpc:=simplify(subs(t=1/x,ehrpc)):
```

```
> openehrhc:=simplify(-subs(t=1/x,ehrhc)):
```

By IOP theory, the generating function for the magic quasipolynomial is

```
> mcgfsum:=openehrpc-openehrhc;
  Mcgfsum:=8*mcgfsum;
```

$$mcgfsum := \frac{x^8}{(x-1)^2\,(x^2-1)\,(x^4-1)} + \frac{x^8}{(x-1)^2\,(x^6-1)}$$

$$Mcgfsum := \frac{8\,x^8}{(x-1)^2\,(x^2-1)\,(x^4-1)} + \frac{8\,x^8}{(x-1)^2\,(x^6-1)}$$

```
> Mcgfs:=simplify(Mcgfsum);
  mcgfs:=simplify(mcgfsum);
```

$$Mcgfs := \frac{8\,x^{10}\,(2\,x^2+1)}{(x^6-1)\,(x^4-1)\,(x-1)^2}$$

$$mcgfs := \frac{x^{10}\,(2\,x^2+1)}{(x^6-1)\,(x^4-1)\,(x-1)^2}$$

Now we bring the total generating function into "normalized form" with denominator "cdenom"...

```
> num:=numer(Mcgfs):
  den:=denom(Mcgfs):
> Magicnum:=expand(simplify(cdenom/den)*num):
  Mcgf:=Magicnum/cdenom:
```

And the symmetry-type generating function ...

```
> num:=numer(mcgfs):
  den:=denom(mcgfs):
> magicnum:=(expand(simplify(cdenom/den)*num)):
  mcgf:=magicnum/cdenom:
```

Expand the series to get all values out to "enddegree".

The comment symbol # should be used to avoid the long computation associated with large enddegree. Only the series being printed should be computed.

```
> #Mcgfseries:=series(Mcgf,x=0,enddegree+1):
  #print("Total series computed.");
> #mcgfseries:=series(mcgf,x=0,enddegree+1):
  #print("Symmetry-type series computed.");
```

List the coefficients of each series, i.e., the terms of the counting sequences.
The comment symbol # should be used for controlling the output. With large "enddegree" the output is huge so it's more convenient to run each sequence's output separately and copy it from the worksheet.

Print the sequence of total counts.

```
> for n from 1 to enddegree do
  #  co:=coeff(Mcgfseries,x,n):
  #  printf("%d  %d \n",n,co);
  od:
  print("Coefficients complete.",n,co);
```

"Coefficients complete.", 101, 39992

Print the sequence of symmetry counts.

```
> for n from 1 to enddegree do
  #  co:=coeff(mcgfseries,x,n):
  #  printf("%d  %d \n",n,co);
  od:
  print("Coefficients complete.",n,co);
```

"Coefficients complete.", 101, 39992

Reduced magic squares: cubic count

First, the generating functions of R_mc(n) and r_mc(n). The conversion factor is:

```
> rmultiplier:=x^2/(1-x)^2:
```

```
> Rgfs:=simplify(Mcgfs/rmultiplier):
  rgfs:=simplify(mcgfs/rmultiplier):
```

Now standardize the denominator. dr = the degree for reduced squares.

```
> dr:=1;
  rdenom:=(1-x^p)^(dr+1);
> num:=numer(Rgfs):
  den:=denom(Rgfs):
> Rmagicnum:=simplify(rdenom/den)*num:
  Rgf:=Rmagicnum/rdenom;
> num:=numer(rgfs):
  den:=denom(rgfs):
> rmagicnum:=simplify(rdenom/den)*num:
  rgf:=rmagicnum/rdenom;
```

$$dr := 1$$

$$rdenom := (1 - x^{12})^2$$

$$Rgf := \frac{8\,(x^{10} + x^8 + x^6 + x^4 + x^2 + 1)\,(x^4 - x^2 + 1)\,(2\,x^2 + 1)\,x^8}{(1 - x^{12})^2}$$

$$rgf := \frac{(x^{10} + x^8 + x^6 + x^4 + x^2 + 1)\,(x^4 - x^2 + 1)\,(2\,x^2 + 1)\,x^8}{(1 - x^{12})^2}$$

Let's generate terms...

```
> Rgfseries:=series(Rgf,x=0,enddegree+1):
  print("Reduced series computed.");
```

"Reduced series computed."

```
> rgfseries:=series(rgf,x=0,enddegree+1):
  print("Reduced symmetry-type series computed.");
```

"Reduced symmetry-type series computed."

List the coefficients of each series, i.e., the terms of the counting sequences.
The comment symbol # should be used for controlling the output. With large "enddegree" the output is huge so it's more convenient to run each sequence's output separately and copy it from the worksheet.

Print the sequence of reduced counts.

```
> for n from 1 to enddegree do
    co:=coeff(Rgfseries,x,n):
    if (modp(n,2)=0) then printf("%d  %d \n",n/2,co); fi:
  od:
  print("Reduced coefficients complete.",n,co);
```

```
1  0
```

```
2   0
3   0
4   8
5   16
6   8
7   24
8   24
9   24
10   32
11   40
12   32
13   48
14   48
15   48
16   56
17   64
18   56
19   72
20   72
21   72
22   80
23   88
24   80
25   96
26   96
27   96
28   104
29   112
30   104
31   120
32   120
33   120
34   128
35   136
36   128
37   144
38   144
39   144
40   152
41   160
42   152
43   168
44   168
45   168
46   176
47   184
48   176
49   192
50   192
```

"Reduced coefficients complete.", 101, 192

Print the sequence of reduced symmetry counts.

```
> for n from 1 to enddegree do
```

```
#  co:=coeff(rgfseries,x,n):
#  if (modp(n,2)=0) then printf("%d  %d \n",n/2,co); fi:
od:
print("Reduced symmetry-class coefficients complete.",n,co);
```

"Reduced symmetry-class coefficients complete.", 101, 192

Magic squares: affine count

The dimension and period for affine magic, and the standard denominator:

```
> da:=2:
  pa:=18:
  adenom:=(1-x^pa)^(da+1):
```

See Section 2.2 of "Six Little Squares". We get LattE to spit out two generating functions--one for the polytope P_a:

```
> magicaffp:= (1-t)/(1-t^3) * (1 / ((1-t) * (1-t^3) * (1-t^6))):
```

... and one for P_a intersect h:

```
> magicaffh:= (1-t)/(1-t^3) * (1 / ((1-t) * (1-t^9))):
```

From this point on we proceed as above...

```
> openmagicaffp:=simplify(-subs(t=1/x,magicaffp)):
```

```
> openmagicaffh:=simplify(subs(t=1/x,magicaffh)):
```

```
> magfsum:=openmagicaffp-openmagicaffh;
  Magfsum:=8*magfsum;
```

$$magfsum := -\frac{x^{12}}{(x^3-1)^2\,(x^6-1)} - \frac{x^{12}}{(x^3-1)\,(x^9-1)}$$

$$Magfsum := -\frac{8\,x^{12}}{(x^3-1)^2\,(x^6-1)} - \frac{8\,x^{12}}{(x^3-1)\,(x^9-1)}$$

```
> Magfs:=simplify(Magfsum);
  magfs:=simplify(magfsum);
```

$$Magfs := -\frac{8\,(2\,x^3+1)\,x^{15}}{(x^9-1)\,(x^3-1)\,(x^6-1)}$$

$$magfs := -\frac{(2\,x^3+1)\,x^{15}}{(x^9-1)\,(x^3-1)\,(x^6-1)}$$

Now we bring the total generating function into "normalized form" with denominator "cdenom"...

```
> num:=numer(Magfs):
  den:=denom(Magfs):
> Magicnumaff:=expand(simplify(adenom/den)*num):
  Magf:=Magicnumaff/adenom:
```

And the symmetry-type generating function ...

```
> num:=numer(magfs):
  den:=denom(magfs):
> magicnum:=(expand(simplify(adenom/den)*num)):
  magf:=magicnum/adenom:
```

Now we expand the series to find the values of the counts, up to "enddegree".

```
> #Magfseries:=series(Magf,x=0,enddegree+1):
  #print("Total series computed.");
```

```
> #magfseries:=series(magf,x=0,enddegree+1):
  #print("Symmetry-class series computed.");
```

List the coefficients of each series, i.e., the terms of the counting sequences.
The comment symbol # is for controlling the output. With large "enddegree" the output is huge so it's more convenient to run each sequence's output separately and copy it from the worksheet.

```
> for n from 1 to enddegree do
  #  co:=coeff(Magfseries,x,n);
  #  if (modp(n,3)=0) then printf("%d  %d \n",n/3,co); fi:
  od:
  print("Total coefficients complete.",n/3,co);
```

$$\text{"Total coefficients complete."}, \frac{101}{3}, 192$$

```
> for n from 1 to enddegree do
  #  co:=coeff(magfseries,x,n):
  #  if (modp(n,3)=0) then printf("%d  %d \n",n,co); fi:
  od:
  print("Coefficients complete.",n,co);
```

"Coefficients complete.", 101, 192

```
>
```