NoBLE for lattice animals in high dimensions:

Using only monotone bounds

Implementation of the computer-assisted proof of the NoBLE by Robert Fitzner and Remco van der Hofstad.

Robert Fitzner
Eindhoven University of Technology

Abstract

This document is the implementation of the computer-assisted proof of the non-backtracking lace expansion (NoBLE). The NoBLE was created by Remco van der Hofstad and the author to prove mean-field behavior for self-avoiding walks, lattice trees (LT), lattice animals (LA) and percolation. In this file the computations for lattice animals are performed. The technique is explained in "Generalized approach to the non-backtracking lace expansion"(I), and the bounds that we implement here are derived in "NoBLE lattice animals and trees"(II). All reference in this file are to one of these two papers, which we refer to by (I) and (II).

This file is accompanied by two other notebooks -SRW.nb- and -General.nb-. In -SRW.nb- a number of simple random walk quantities are computed. In -General.nb- general bounds, derived in (I), are implemented. Before doing computations with this file, the user should first open these files, choose a dimension and once execute all lines of the file.

In this file we use only bounds for which we can prove that they are monotone decreasing in the dimension, see Appendix B of (II). In this way any successful analysis in a given dimension d' yields that the analysis will also be successful for all d≥d'.

The computation of the -SRW.nb- and -General.nb- file are independent of the values Γ_i and c, so that we need to execute these files only once, when you start the Mathematica Kernel or when you want to change the dimension under consideration.

20.03.2019

Initial parameters

We start by defining the bounds assumed for the bootstrap Γ_i, c_μ and $c_{n,l,S}$. For the following values the bootstrap succeeds in dimension 30:

```
In[4163]:= (*The parameter choices that work in d=30*)
Gamma1 = 1.06; (*Assumed bound on zg_z*)
Gamma2 = 1.125; (*(2d-1)/(2d-2) sup_k [1-D̂(k)] Ĝ_z(k)*)
Gamma3 = 1;
cmu = 1.033;
GammaThree[1, 0] = 0.05;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) G(x-y)*)
GammaThree[1, 1] = 0.007;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D*G) (x-y)*)
GammaThree[1, 2] = 0.003; (*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D*D*G) (x-y)*)
GammaThree[1, 3] = 0.001; (*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D^{*3}*G) (x-y)*)
GammaThree[2, 0] = 0.7;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (G*G) (x-y)*)
GammaThree[2, 1] = 0.016;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D*G*G) (x-y)*)
GammaThree[2, 2] = 0.005;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D*D*G*G) (x-y)*)
GammaThree[2, 3] = 0.002;(*Assumed bound on sup_{x≠0} ∑_y |y|_2^2 G(y) (D^{*3}*G*G) (x-y)*)
GammaThreeClosed[1, 4] = 0.005;(*Assumed bound on ∑_y |y|_2^2 G(y) (D^{*6}*G) (-y)*)
GammaThreeClosed[2, 4] = 0.012;
(*Assumed bound on ∑_y |y|_2^2 G(y) (D^{*6}*G*G) (-y)*)
```

Loading the allowed values

Now we prepare all bounds on SRW integrals, which are proven to be monotone in the dimension. Only these are used in the later parts of the notebook.

$I_{n,m}(x)$

In[4177]:=
```
validPoints = {{0}, {1}, {2}, {3}, {4}, {0, 1}, {1, 1}, {0, 2}, {1, 2}, {2, 1}};
Do[
  Do[
    Inm[v, n, 0] = SRWIntegral[v, n, d];

    Do[
     Inm[v, n, l] = Inm[v, n, 0];
     , {l, 1, 10}]; (*Fill them with initial value. The important ones
     will be replaced below*)
    , {v, validPoints}];
  , {n, 1, param}];
Do[Do[
    Inm[v, 0, l] = SRWTo[v, l, d] / (2 d)^l;
    , {l, 0, 10}]; (*Fill them with initial value. The important ones
   will be replaced below*)
  , {v, validPoints}];

Do[
  Do[
   Inm[{0}, n, l] = Inm[{1}, n, l - 1];
   Inm[{1}, n, l] =
    1/(2 d) (Inm[{0}, n, l - 1] + Inm[{0, 1}, n, l - 1] + 2 d Inm[{2}, n, l - 1]);
   Inm[{2}, n, l] =
    1/(2 d) (2 Inm[{1}, n, l - 1] + 2 Inm[{1, 1}, n, l - 1] + 2 d Inm[{3}, n, l - 1]);
   Inm[{3}, n, l] =
    1/(2 d) (3 Inm[{2}, n, l - 1] + 3 Inm[{2, 1}, n, l - 1] + 2 d Inm[{4}, n, l - 1]);
   Inm[{0, 1}, n, l] =
    1/(2 d) (Inm[{1}, n, l - 1] + Inm[{0, 1}, n, l - 1] + 2 d Inm[{1, 1}, n, l - 1]);
   Inm[{0, 2}, n, l] =
    1/(2 d) (2 Inm[{1, 1}, n, l - 1] + 2 Inm[{0, 2}, n, l - 1] + 2 d Inm[{1, 2}, n, l - 1]);
   Inm[{1, 1}, n, l] =
    1/(2 d) (Inm[{2}, n, l - 1] + Inm[{1, 1}, n, l - 1] + Inm[{0, 1}, n, l - 1] +
       Inm[{0, 2}, n, l - 1] + 2 d Inm[{2, 1}, n, l - 1]);
   Inm[{2, 1}, n, l] =
    1/(2 d) (Inm[{3}, n, l - 1] + Inm[{2, 1}, n, l - 1] + 2 Inm[{1, 1}, n, l - 1] +
       2 Inm[{1, 2}, n, l - 1] + 2 d Inm[{2, 1}, n, l - 1]);
   , {l, 1, 10}]
  , {n, 1, param}];
```

Now we use the computed values of $\bar{I}_{n,l}(x)$ to bound the other SRW integrals, see Section 5.2. We begin with $L_n(x)$ which we can compute explicitly, see (5.7) and (5.17).

$L_n(x)$

```
In[4181]:= Do[
    L[n, {0}] = Inm[{0}, n, 0];
    L[n, {1}] = 1/(2 d) (Inm[{0}, n, 0] + Inm[{0, 1}, n, 0] + 2 d Inm[{2}, n, 0]);
    L[n, {2}] = Inm[{4}, n, 1] + 1/(2 d) (Inm[{2}, n, 1] + Inm[{2, 1}, n, 1]) +
      1/(4 d (d - 1)) (Inm[{0}, n, 0] + Inm[{0, 2}, n, 0] + 2 Inm[{0, 1}, n, 0]);
    L[n, {0, 1}] = 1/(2 d) (Inm[{0}, n, 0] + Inm[{0, 1}, n, 0] + 2 d Inm[{0, 2}, n, 0]);
    , {n, 0, param}];
  LnPoints = {{0}, {1}, {0, 1}, {2}};
```

$K_{n,l}(x)$

Then we bound $K_{n,l}(x)$ as described in (3.36), and then improve the bound for $x = e_1$, $e_1 + e_2$, $2\,e_1$.

```
In[4183]:= Do[
    Do[
     Do[
      K[n, l, v] = Abs[Inm[{0}, n, 2 l]^(1/2) L[n, v]^(1/2)];
      , {v, LnPoints}];
     K[n, l, {1}] = Min[Abs[Inm[{0}, n, 2 l]^(1/2) L[n, {1}]^(1/2)],
       Abs[Inm[{0}, n, l + 1]]];
     , {l, 0, 5}];
    , {n, 0, param}];
```

For l=0 we can improve this to:

```
In[4184]:= Do[
   Do[
    K[n, 0, v] = Abs[K[n - 1, 0, v] + (Inm[{0}, n - 1, 2])^(1/2) (L[n - 1, v])^(1/2) +
       Inm[{0}, n, 4]^(1/2) (L[n, v])^(1/2)];
    , {n, 1, param}], {v, {{1}, {0, 1}, {2}}}]
```

For $x = 0$ and even m we use the slightly better bound (5.14):

```
In[4185]:= Do[
    Do[
     K[n, 2 l, {0}] = Inm[{0}, n, 2 l];
     , {l, 0, 5}];
    , {n, 0, param}];
```

$T_{n,l}(x)$

We bound $T_{n,l}(x)$ as defined in (3.37) and bounded in (5.11):

```
In[4186]:= Do[Do[Do[
        T[n, l, v] = K[n, l + 1, v] + Min[(4/d) K[n, l, v], (2/d) K[n + 1, l, v]];
        , {n, 1, param - 1}];
      T[param, l, v] = K[param, l + 1, v] + (4/d) K[param, l, v];
      , {l, 0, 4}];

    Do[Do[
        T[n, l, v] = T[n, 4, v]
        , {l, 5, 8}];, {n, 1, param}];
    , {v, LnPoints}];
```

$V_{n,l}$

We compute the bounds on $V_{n,l}$ defined in (3.37) and (5.8), and computed in (5.25):

```
In[4187]:= Do[
   Do[
     V[n, l] =
       1/(2 d)^2 (Inm[{0}, n, l] + Inm[{0, 2}, n, l] + 1/(2 d) Inm[{0}, n, l] +
          1/(2 d) Inm[{0, 1}, n, l]);
     , {l, 0, 8}];
   , {n, 0, param}]
```

Note that we cannot subtract $\bar{I}_{n,m}(2\, e_1)$ as we cannot guarantee that the result is monotone in the dimension.

$U_{n,l}(x)$

Then we compute $U_{n,l}(x)$ as given in (3.38) and bounded in (5.9) and (5.12)-(5.13):

```
In[4188]:= Do[Do[Do[
        U[n, l, v] = Min[V[n, 2 l]^(1/2) L[n, v]^(1/2), (1/d) K[n, l, v]];
        , {v, LnPoints}];
      , {n, 0, param}]
    , {l, 0, 4}];
   Do[Do[Do[
        U[n, l, v] = Min[V[n, 8]^(1/2) L[n, v]^(1/2), (1/d) K[n, 5, v]];
        , {v, LnPoints}];
      , {n, 0, param}]
    , {l, 5, 6}];
```

Weighted SRW-diagrams

We begin with the bound for the initial point, derived in Section 3.3.3, see (3.29).

$$J_{n,l}(x) = I_{n+2,l+1}(x) - \frac{1}{d} I_{n+3,l}(x) + \frac{1}{2 d^2} \Sigma_\iota I_{n+3,l}(x + 2 e_i)$$

In[4190]:=
```
Jnl[{0}, n_, l_] := Inm[{0}, n + 2, l + 1] + 1/d Inm[{0, 1}, n + 3, l];
Jnl[{1}, n_, l_] :=
  Inm[{1}, n + 2, l + 1] +
   1/(2 d^2) (Inm[{1}, n + 3, l] + Inm[{0, 1}, n + 3, l] + 2 d Inm[{1, 1}, n + 3, l]);
Jnl[{2}, n_, l_] :=
  Inm[{2}, n + 2, l + 1] +
   1/(2 d^2) (2 Inm[{2}, n + 3, l] + 2 Inm[{1, 1}, n + 3, l] + 2 d Inm[{2, 1}, n + 3, l]);
Jnl[{0, 1}, n_, l_] :=
  Inm[{0, 1}, n + 2, l + 1] +
   1/(2 d^2) (Inm[{0}, n + 3, l] + Inm[{0, 1}, n + 3, l] + 2 d Inm[{0, 2}, n + 3, l]);
```

Bound on the two-point function and on repulsive diagrams

In this section we use the bootstrap assumption $f_i(z) < \Gamma_i$ and the computation of -SRW.nb- to conclude bounds on the two-point function and the basic diagrams

Bound on z and g_z

We define the constants for two settings indicated by s: we use s=i for bound on $z = z_I$ and s=o for bound on $z \epsilon (z_I, z_c)$: For $z = z_I$, we use

$$z_I = \frac{1}{(2 d - 1) e} \qquad g_{z_I} \le e + \frac{e - 1}{2 d - 1} \qquad g^i_{z_I} \le 1 + (g_{z_I} - 1) \frac{2 d - 1}{2 d} \le e \tag{1}$$
$$\overline{G}_z(x) \le g^i_{z_I} B_{z_I\, g^i_{z_I}}(x) \le g^i_{z_I} C(x) \qquad \tilde{G}_z(0) = 1 \text{ and otherwise } \tilde{G}_z(x) \le B_{z_I\, g^i_{z_I}}(x) \le C(x).$$

and implement this as:

In[4194]:=
```
z[i] = 1/((2 d - 1) Exp[1]);
gj[i] = Exp[1];
g[i] = Exp[1] + (Exp[1] - 1)/(2 d - 1);
gz[i] = g[i] × z[i];
gjz[i] = gj[i] × z[i];
VarGamma2[i] = 1;
VarGamma2b[i] = VarGamma2[i] gz[i]/gjz[i];
```

For the $z \epsilon (z_I, z_c)$ we assume that

$$2 d z g^\iota_z < 2 d g_z z < \Gamma_1, \quad g_z < e \Gamma_1 \text{ for all } z \ge \frac{1}{(2 d - 1) e}, \quad g^\iota_z < 1 + (g_z - 1) \frac{2 d - 1}{2 d}. \tag{2}$$

In[4201]:=
```
gz[o] = Gamma1/(2 d - 1);
gjz[o] = 1/cmu Gamma1/(2 d - 1);
g[o] = Exp[1] Gamma1;
gj[o] = Min[Exp[1] cmu Gamma1, 1 + (g[o] - 1) (2 d - 1)/(2 d)];
VarGamma2[o] = Gamma2;
VarGamma2b[o] = VarGamma2[o] gz[o]/gjz[o];
```

Lower bound on g_z

For some bounds, we require a bound on $z < z_c$. For this we create a lower bound on g_z, which we obtain by using that we only consider $z > z_I = \frac{1}{(2d-1)e}$ and the following basic bounds:

$$\begin{gathered} g_z = \Sigma_{n=0} t_n(0) z^n \geq \Sigma_{n=0}^{3} (\text{lower bound on } t_n(0)) z_I^n \\ (n=0) \implies t_n(0) z_I^n = 1 \\ (n=1) \implies t_1(0) z_I^1 = 2d * \frac{1}{(2d-1)e} > \frac{1}{e} \\ (n=1) \implies t_2(0) z_I^2 = 3d(2d-1) * \frac{1}{(2d-1)^2 e^2} = \frac{3}{2} \frac{d}{d-\frac{1}{2}} \frac{1}{e^2} > \frac{1.5}{e^2} \end{gathered} \tag{3}$$

In[4207]:=
```
gLower = 1 + 1/Exp[1] + 1.5/Exp[2];
"Which is " <> TextString[NumberForm[N[gLower/g[o] * 100], 5]] <>
  "% of the upper bounds."
z[o] = gz[o]/gLower;
```

Out[4208]=
```
Which is 54.518% of the upper bounds.
```

Bounds on two-point functions

Here, we compute bounds on the two-point functions for some x, as explained in Section 5.3.2. of (I). These bounds are obtained by extracting short, explicit contributions and by bounding the longer contributions using f_2, see Section 5.3 of (I). To bound the short explicit contributions we use the values $c_j(x)$ =nrSAW[j,d,x] provided in the model-independent SRW-integral notebook, these are provided for *j≤ComputedSteps*. The value of *ComputedSteps* is given in the -SRW.nb- file. The extraction of short contributions creates a better bound than just applying f_2. The reason is that f_1 gives a sharper bound than f_2.

$$\tilde{G}_{m,z}(x) \leq \sum_{j=m}^{M-1} (g_z^{\iota} z)^j c_j(x) + \tilde{G}_{M,z}(x). \tag{4}$$

For lattice animals, we have in total defined three different two-point functions, namely $G_{m,z}, \overline{G}_{m,z}, \tilde{G}_{m,z}$. The function $G_{m,z}, \overline{G}_{m,z}$ can be bounded directly using f_2. For $\tilde{G}_{m,z}$ we use an additional bound

$$\tilde{G}_{m,z}(x) \leq 2dz\left(D * \overline{G}_{m-1,z}\right)(x) = 2dz g_z (D * G_{m-1,z})(x) \leq 2d g_z \left(2dz g_z^{\iota}\right)^{m-1} (D^m * G_z)(x) \leq 2d g_z \left(2dz g_z^{\iota}\right)^{m-1} \Gamma_2 K_{1,m}(x). \tag{5}$$

for even m. We use the following variables, in which m is required to be even:

Bound on $G_{m,z}(e_1)$

We use $G_{m,z}(e_1) = (D * G_{m,z})(0)$ to compute

In[4210]:=
```
Do[
 Do[
  Bound[tG, {1}, m, s] = Sum[gjz[s]^j nrSAW[j, d, {1}], {j, m, 5}] +
     2 d gz[s] (2 d gjz[s])^6 VarGamma2[s] × Inm[{0}, 1, 8];
  , {m, 1, 5}]
 , {s, {i, o}}]
```

Bound on $G_{m,z}(e_1 + e_2)$

We use $G_{m,z}(e_1 + e_2) \leq \frac{d}{d-1} \left(D^2 * G_{m,z}\right)(0)$ to compute

In[4211]:=
```
Do[
  Do[
   Bound[tG, {2}, 2 m, s] = Sum[gjz[s]^j nrSAW[j, d, {2}], {j, 2 m, 4}] +
      d/(d - 1) 2 d gz[s] (2 d gjz[s])^5 VarGamma2[s] × Inm[{0}, 1, 8];
   , {m, 1, 3}], {s, {i, o}}];
```

Bound on $G_{m,z}(2\,e_1)$

Now, we compute bounds on $G_{m,z}(2\,e_1)$.

In[4212]:=
```
Do[
  Do[
   Bound[tG, {0, 1}, 2 m, s] = Sum[gjz[s]^j nrSAW[j, d, {0, 1}], {j, 2 m, 4}] +
     2 d gz[s] (2 d gjz[s])^5 VarGamma2[s] × K[1, 4, {0, 1}];
   , {m, 1, 2}], {s, {i, o}}];
```

Bound on $\sup_{x\neq 0} G_{m,z}(x)$

To compute the supremum of the two-point function we use that $c_n(x)$=nrSAW[n,d,x] for $n \leq 10$ has its maximal value at $x = e_1$ or $x = e_1 + e_2$.

In[4213]:=
```
Do[
 Bound[tG, max, 4, s] = Max[Bound[tG, {1}, 5, s], Bound[tG, {2}, 4, s],
   Bound[tG, {0, 1}, 4, s] ];
 Bound[tG, max, 3, s] = Max[Bound[tG, {1}, 3, s], Bound[tG, {2}, 4, s],
   Bound[tG, {0, 1}, 4, s] ];
 Bound[tG, max, 2, s] = Max[Bound[tG, {1}, 3, s], Bound[tG, {2}, 2, s],
   Bound[tG, {0, 1}, 2, s] ];
 Bound[tG, max, 1, s] = Max[Bound[tG, {1}, 1, s], Bound[tG, {2}, 2, s],
   Bound[tG, {0, 1}, 2, s] ];
 Bound[tG, max, 0, s] = Bound[tG, max, 1, s];
 , {s, {i, o}}]
```

Bound on $\frac{g_z}{g_z'}$ and $\frac{g_z'}{g_z}$

In[4214]:=
```
Do[
 mubOverMu[s] = 1 + Bound[tG, {1}, 1, s];
 muOverMub[s] = 1;
 , {s, {i, o}}]
```

Repulsive diagrams

Now, we bound repulsive diagrams, defined in Definition 4.7 of (II) as described in Section 5.3.2 of (I). In (i) we see that the resulting bounds do no depend on the individual lengths of the pieces $m_1, m_2, \ldots$, but only of the sum of the known length $\sum_i m_i$. So we refer to each diagram by the number of minimal steps $\sum_i m_i$ and the number of two-point function without fixed length ($G_{m,z}(x)$ instead of $G_{\underline{m},z}(x)$). Further, we assume that all only two-point functions $\tilde{G}_{m,z}$ are involved.

Closed diagrams

```
In[4215]:= Do[
  Bound[Loop, 4, s] = 2 d gjz[s] × Bound[tG, {1}, 3, s];
  Do[
   Bound[Bubble, m, s] = Sum[(j + 1 - m) 2 d nrBAW[j - 1, d, {1}] gjz[s]^j, {j, m, 7}] +
     (8 - m) (2 d gjz[s])^8 VarGamma2b[s] × Inm[{0}, 1, 8] +
     (2 d gjz[s])^8 VarGamma2b[s]^2 Inm[{0}, 1, 8]; (* (5.40) of I*)
   , {m, 2, 6}];
  Bound[Double, 2, s] = Bound[Loop, 4, s] + Bound[Bubble, 4, s]/2;
  Bound[Double, 4, s] = Bound[Bubble, 4, s]/2;
  Do[
   Bound[Triangle, m, s] =
    Sum[((j + 1 - m) (j + 2 - m))/2 2 d nrBAW[j - 1, d, {1}] gjz[s]^j, {j, m, 7}] +

     ((8 - m) × (8 - 1 - m))/2 (2 d gjz[s])^8 VarGamma2b[s] × Inm[{0}, 1, 8] +
     (8 + 1 - m) (2 d gjz[s])^8 VarGamma2b[s]^2 Inm[{0}, 2, 8] +
     (2 d gjz[s])^8 VarGamma2b[s]^3 Inm[{0}, 3, 8]; (* (5.41) of I*)
   , {m, 1, 5}];
  Do[
   Bound[Square, m, s] =
    Sum[((j + 1 - m) (j + 2 - m) (j + 3 - m))/6 2 d nrBAW[j - 1, d, {1}] gjz[s]^j, {j, m, 7}] +
     1/6 × (8 + 1 - m) × (8 + 2 - m) × (8 + 3 - m) (2 d gjz[s])^8 VarGamma2b[s] × Inm[{0}, 1, 8] +
     ((8 - m) × (8 - 1 - m))/2 (2 d gjz[s])^8 VarGamma2b[s]^2 Inm[{0}, 2, 8] +
     (8 - m) (2 d gjz[s])^8 VarGamma2b[s]^3 Inm[{0}, 3, 8] +
     gz[s]/gjz[s] (2 d gjz[s])^8 VarGamma2b[s]^4 Inm[{0}, 4, 8]; (* (5.42) of I*)
   , {m, 2, 5}];
  , {s, {i, o}}]
```

Open repulsive diagrams

We can bound open repulsive diagrams, in the same way as the closed diagrams. Parallel to this we also produce the bounds without extracting contributions, as it is a priori not clear which bound is better. We use the monotonicity of the SRW-integrals, see Lemma 5.1 of (I), to conclude that the supremums of the open diagrams is at the neighbor of the origin.

```
In[4216]:= Do[
  Do[
   extracted = 3;
   Bound[OpenBubblePre, m, s] =
    Min[gz[s]/gjz[s] (2 d gjz[s])^m VarGamma2[s]^2 K[2, m, {1}],
     Max[Sum[(j + 1 - m) gjz[s]^j nrBAW[j, d, {2}], {j, m, extracted}],
       Sum[(j + 1 - m) gjz[s]^j nrBAW[j, d, {1}], {j, m, extracted}]]
      + (extracted + 1 - m) (2 d gjz[s])^(extracted+1) VarGamma2b[s] ×
       K[1, extracted + 1, {1}] + (2 d gjz[s])^(extracted+1) VarGamma2b[s]^2
       K[2, extracted + 1, {1}]];
   extracted = 4;
   Bound[OpenTrianglePre, m, s] =
    Min[(2 d gjz[s])^m VarGamma2b[s]^3 K[3, m, {1}],
     Max[Sum[((j + 1 - m) (j + 2 - m))/2 gjz[s]^j nrBAW[j, d, {2}], {j, m, extracted}],
       Sum[((j + 1 - m) (j + 2 - m))/2 gjz[s]^j nrBAW[j, d, {1}], {j, m, extracted}]] +
      1/2 (extracted + 1 - m) (extracted + 1 - 1 - m) (2 d gjz[s])^(extracted+1)
       VarGamma2b[s] × K[1, extracted + 1, {1}] +
      (extracted + 1 - m) (2 d gjz[s])^(extracted+1) VarGamma2b[s]^2
       K[2, extracted + 1, {1}] + (2 d gjz[s])^(extracted+1) VarGamma2b[s]^3
       K[3, extracted + 1, {1}]];
   Bound[OpenSquarePre, m, s] =
    Min[(2 d gjz[s])^m VarGamma2b[s]^4 K[4, m, {1}],
     Max[Sum[1/6 (j + 1 - m) (j + 2 - m) (j + 3 - m) gjz[s]^j nrBAW[j, d, {2}],
        {j, m, extracted}],
       Sum[1/6 (j + 1 - m) (j + 2 - m) (j + 3 - m) gjz[s]^j nrBAW[j, d, {1}],
        {j, m, extracted}]] +
      1/6 × (10 + 1 - m) × (10 + 2 - m) × (10 + 3 - m) (2 d gjz[s])^(extracted+1)
       VarGamma2b[s] × K[1, extracted + 1, {1}] +
      ((10 - m) × (10 - 1 - m))/2 (2 d gjz[s])^(extracted+1) VarGamma2b[s]^2
       K[2, extracted + 1, {1}] + (10 - m) (2 d gjz[s])^(extracted+1) VarGamma2b[s]^3
       K[3, extracted + 1, {1}] + (2 d gjz[s])^(extracted+1) VarGamma2b[s]^4
       K[4, extracted + 1, {1}]];
   , {m, 0, 4}];
  , {s, {i, o}}]
```

When trying to find an optimal bound we saw that the following idea improved the bounds slightly. Let us show the idea in the example of a repulsive triangle with some length restrictions:

$$T_{1,0,2}(x) \le T_{\underline{1},0,2}(x) + T_{2,0,2}(x) \le T_{\underline{1},0,2}(x) + T_{\underline{2},0,2}(x) + T_{\underline{3},0,2}(x) + T_{4,0,2}(x)$$

In[4217]:=
```
Do[
  m = 4;
  Bound[OpenBubble, m, s] = Bound[OpenBubblePre, m, s];
  Bound[OpenTriangle, m, s] = Bound[OpenTrianglePre, m, s];
  Bound[OpenSquare, m, s] = Bound[OpenSquarePre, m, s];
  Clear[m];
  , {s, {i, o}}];
```

Using this initial step we iteratively consider smaller and smaller diagrams and in each step use the best bound possible:

In[4218]:=
```
Do[
  Do[
    Bound[OpenBubble, m, s] = Min[Bound[OpenBubblePre, m, s],
      Bound[tG, max, m, s] + Bound[OpenBubble, m + 1, s]];
    Bound[OpenTriangle, m, s] = Min[Bound[OpenTrianglePre, m, s],
      Bound[OpenBubble, m, s] + Bound[OpenTriangle, m + 1, s]];
    Bound[OpenSquare, m, s] = Min[Bound[OpenSquarePre, m, s],
      Bound[OpenTriangle, m, s] + Bound[OpenSquare, m + 1, s]];
    , {m, {3, 2, 1, 0}}];
  , {s, {i, o}}];
```

Weighted Diagrams

Here we define the bounds on the weighted diagrams. As explained in Section 5.3.3 of (I) we use for $z = z_I$ a bound that is independent of our analysis. These bounds have been implemented in the accompanying notebook. To the notation: As for the unweighted diagrams, we refer to the diagrams by their number of two-point functions and number of the fixed steps on the unweighted lines. First we replace a function, that is given in the general notebook, to ensure that only monotone bounds are used.

In[4219]:=
```
BoundFThreeInital[d_, n_, l_, rho_, vecs_] := Module[{v2, i},
   v2 = Table[rho ((2 d - 2)/(2 d - 1))^(n+1) Jnl[vecs[[i]], n, l], {i, 1, Length[vecs]}];
   Max[v2]
  ];
```

In[4220]:=
```
Bound[WeightedBubble, 4, i] = (2 d gjz[i])^4 BoundFThreeInital[d, 1, 4, 1, {{0}}] ;
Bound[WeightedTriangle, 4, i] = (2 d gjz[i])^4 BoundFThreeInital[d, 2, 4, 1, {{0}}] ;

Do[
 Bound[WeightedOpenBubble, t, i] =
  (2 d gjz[i])^t BoundFThreeInital[d, 1, t, 1, {{1}}];
 Bound[WeightedOpenTriangle, t, i] =
  (2 d gjz[i])^t BoundFThreeInital[d, 2, t, 1, {{1}}];
 , {t, 0, 3}]
```

For $z \in (z_I, z_c)$ we use the bootstrap function f_3 to obtain the bonds

```
In[4223]:= Bound[WeightedBubble, 4, o] = (2 d gjz[o])^4 GammaThreeClosed[1, 4];
Bound[WeightedTriangle, 4, o] = (2 d gjz[o])^4 GammaThreeClosed[2, 4];
Do[
 Bound[WeightedOpenBubble, t, o] = (2 d gjz[o])^t GammaThree[1, t];
 Bound[WeightedOpenTriangle, t, o] = (2 d gjz[o])^t GammaThree[2, t];
 , {t, 0, 3}]
```

As explained in Section 5.3.3, we drastically improve the bounds on the closed, weighted, repulsive diagram by extracting explicit contributions, by using its repulsiveness and

$$\frac{1}{g_z} \sum_x \| x \|_2^2 \sum_{A,\, x \epsilon \mathrm{A}} z^A \leq \frac{1}{g_z} \sum_x \| x \|_2^2 \, \overline{G}_z(x) \, 2\, d\, z \left(D * \tilde{G}\right)(x) \leq \sum_x \| x \|_2^2 \, G_z(x) \, 2\, d\, z \left(D * \tilde{G}\right)(x)$$

$$\frac{1}{g_z} \sum_x \| x \|_2^2 \sum_{A,\, x \epsilon \mathrm{A}} 1_{d(0,x) > n} \, z^A \leq \sum_x \| x \|_2^2 \, G_z(x) \frac{\left(2\, d\, z\, g_z^\iota\right)^n}{g_z^\iota} \left(D^{*n} * \tilde{G}\right)(x)$$

We bound the unweighted connections $\left(G^{*n} * D^{*l}\right)(x)$ using $K_{n,l}$ (x), defined in (3.36) of (I). We compute $K_{n,l}(x)$ for $l \leq rem$, see below, and some points close to the origin. For x for which we have not computed $K_{n,l}(x)$ we use a monotonicity argument to bound $K_{n,l}(x)$, in our case `K (2 e1) > K (x)` for all relevant x. This monotonicity is implied by Lemma 5.1 of (I).

In[4226]:=
```
Do[
 explicit = 3;
 LongContributions[point_] = (2 d gz[s]) (2 d gjz[s])^3 K[4, 4, {1}];

 Bound[WeightedBubble, 3, s] =
  Bound[WeightedBubble, 4, s] +
   1/gLower
    (gjz[s]^3 nrSAW[3, d, {1}] *
       (Sum[ nrBAW[r, d, {1}] gjz[s]^r, {r, 1, explicit}] + LongContributions[{1}]) +
      5 gjz[s]^3 nrSAW[3, d, {1, 1}] *
       (Sum[ nrBAW[r, d, {1, 1}] gjz[s]^r, {r, 1, explicit}] +
         LongContributions[{0, 1}]) +
      9 gjz[s]^3 nrSAW[3, d, {0, 0, 1}] *
       (Sum[ nrBAW[r, d, {0, 0, 1}] gjz[s]^r, {r, 1, explicit}] +
         LongContributions[{0, 1}]) +
      3 gjz[s]^3 nrSAW[3, d, {3}] *
       (Sum[ nrBAW[r, d, {3}] gjz[s]^r, {r, 1, explicit}] +
         LongContributions[{2}]));

 Bound[WeightedBubble, 2, s] =
  Bound[WeightedBubble, 3, s] +
   1/gLower (2 d * 4 * gjz[s]^2 Bound[tG, {0, 1}, 4, s] +
      2 * 2 * gjz[s]^2 2 d (2 d - 2) Bound[tG, {2}, 2, s]);
 Bound[WeightedBubble, 1, s] =
  Bound[WeightedBubble, 2, s] + 2 d z[s] × Bound[tG, {1}, 3, s];
 Bound[WeightedBubble, 0, s] = Bound[WeightedBubble, 1, s];

 Bound[WeightedTriangle, 3, s] =
  Bound[WeightedBubble, 3, s] + Bound[WeightedTriangle, 4, s];
 Bound[WeightedTriangle, 2, s] =
  Bound[WeightedBubble, 2, s] + Bound[WeightedTriangle, 3, s];
 Bound[WeightedTriangle, 1, s] =
  Bound[WeightedBubble, 1, s] + Bound[WeightedTriangle, 2, s];
 Bound[WeightedTriangle, 0, s] =
  Bound[WeightedBubble, 0, s] + Bound[WeightedTriangle, 1, s];
 (*remove the auxiliary variables from the memory*)
 Clear[LongContributions];

 Bound[WeightedDouble, 2, s] = Bound[WeightedBubble, 2, s]/2;
 Bound[WeightedDouble, 1, s] =
  Bound[WeightedDouble, 2, s] + 2 d z[s] × Bound[tG, {1}, 3, s];
 , {s, {i, o}}]
```

Building Blocks

Blocks without weight

In the following we implement the bound on the coefficient $A^{m,l}$ defined in Appendix C.1 of (II).

In[4227]:=
```
Do[
  Bound[A, 0, 0, s] = Bound[Loop, 4, s] + 2 Bound[Bubble, 3, s] +
    Bound[Triangle, 3, s];
  (*Table 2*)
  Bound[A, 0, 1, s] = Bound[Loop, 4, s] + 2 Bound[Bubble, 3, s] +
    Bound[Triangle, 4, s];
  (*Table 3*)
  Bound[A, 0, 2, s] = Bound[Bubble, 3, s] + 2 Bound[Triangle, 4, s] +
    Bound[Square, 5, s];
  (*Table 4*)
  Bound[A, 0, -1, s] = Bound[Loop, 4, s] (*d0,x=1,w=y*) + Bound[Bubble, 4, s]
    (*d0,x=1,0≠w≠y*) + Bound[Triangle, 3, s] (*d0,x>2*); (*Table 5*)
  Bound[A, 0, -2, s] = Bound[Triangle, 4, s] (*d0,x=1→w≠0*) +
    Bound[Triangle, 4, s] (*d0,x≥2,w=0*) + Bound[Square, 5, s] (*d0,x≥2,w≠0*);
  (*Table 6*)



  Bound[A, -1, 0, s] = Bound[Bubble, 3, s]/(2 d gjz[s]) + Bound[Triangle, 3, s]/(2 d gjz[s]);
  Bound[A, -1, -1, s] = Bound[Triangle, 3, s]/(2 d gjz[s]);
  Bound[A, -1, -2, s] = Bound[Square, 4, s]/(2 d gjz[s]);
  Bound[A, -1, 1, s] = Bound[Triangle, 3, s]/(2 d gjz[s]);
  Bound[A, -1, 2, s] = Bound[Square, 4, s]/(2 d gjz[s]);

  Bound[A, -2, 0, s] = Bound[OpenBubble, 2, s] + Bound[OpenTriangle, 2, s];
  Bound[A, -2, 1, s] = Bound[OpenTriangle, 2, s];
  Bound[A, -2, -1, s] = Bound[OpenTriangle, 2, s];
  Bound[A, -2, 2, s] = Bound[OpenSquare, 3, s];
  Bound[A, -2, -2, s] = Bound[OpenSquare, 3, s];
  Do[Do[
    Bound[A, a, b, s] = Bound[A, -a, b, s];
    , {a, {1, 2}}], {b, {-2, -1, 0, 1, 2}}];

  Do[
   Bound[Abar, b, 0, s] = Bound[A, b, 0, s];
   Bound[Abar, 0, b, s] = Bound[Abar, b, 0, s]
   , {b, -2, 2}];
```

```
Do[
 Bound[Abar, b, 1, s] = Bound[Abar, 0, 1, s] / (2 d gjz[s]);
 Bound[Abar, b, -1, s] = Bound[Abar, 0, -1, s] / (2 d gjz[s]);
 , {b, {-1, 1}}];

Do[Do[
  Bound[Abar, a, b, s] = (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 1, {0}];
  , {a, {-2, 2}}], {b, {-2, 2}}];

Do[Do[
  Bound[Abar, a, b, s] = (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 2, {1}];
  Bound[Abar, b, a, s] = (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 2, {1}];
  , {a, {-1, 1}}], {b, {-2, 2}}];
, {s, {i, o}}];
```

Blocks with weight

Here we implement the element stated in Appendix C.2 of (II).

```
In[4228]:= Do[
Bound[C, 0, 0, s] = 2 Bound[WeightedBubble, 2, s] + Bound[WeightedTriangle, 2, s];
Bound[C, 0, -1, s] = Bound[WeightedBubble, 2, s] / (2 d gjz[s]) + Bound[WeightedTriangle, 3, s] / (2 d gjz[s]);
Bound[C, -1, 0, s] = Bound[C, 0, -1, s];
Bound[C, 0, 1, s] = Bound[C, 0, -1, s];
Bound[C, 1, 0, s] = (*d0,x=1,v=w≠ x*)
 (4 * gjz[s] × Bound[tG, {0, 1}, 4, s] + 2 * gjz[s] (2 d - 2) Bound[tG, {2}, 2, s]) +
  (*d0,x=1,v≠w≠ x,split weight*) 2 Bound[WeightedBubble, 3, s] / (2 d gjz[s]) +
  (*d0,x≥ 2,split weight*) 2 Bound[WeightedTriangle, 3, s] / (2 d gjz[s]);
Bound[C, -1, -1, s] = (*v=w=y*)
 Max[4 * Bound[tG, {0, 1}, 4, s], 2 * Bound[tG, {2}, 2, s]] + (*v≠w=y or v=w≠y*)
  2 Bound[WeightedBubble, 3, s] / (2 d gjz[s])^2 + (*v≠w≠y*) Bound[WeightedTriangle, 4, s] / (2 d gjz[s])^2;
Bound[C, -1, 1, s] = Bound[C, -1, -1, s];
Bound[C, 1, -1, s] = 2 Bound[WeightedBubble, 3, s] / (2 d gjz[s])^2 +
  4 Bound[WeightedTriangle, 4, s] / (2 d gjz[s])^2;
Bound[C, 1, 1, s] = Bound[C, 1, -1, s];
(*the bad bounds*)
Bound[C, 0, 2, s] = Bound[WeightedOpenTriangle, 0, s];
```

```
Bound[C, 0, -2, s] = Bound[WeightedOpenTriangle, 0, s];
Do[
 Bound[C, -2, t, s] = Bound[WeightedOpenTriangle, 0, s];
 Bound[C, 2, t, s] = 2 Bound[WeightedOpenTriangle, 0, s];
 , {t, {-2, 0, 2}}];
```

$$\text{Bound[C, -2, 1, s]} = \frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, -2, -1, s]} = \frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, -1, 2, s]} = \frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, -1, -2, s]} = \frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, 2, 1, s]} = 2\,\frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, 2, -1, s]} = 2\,\frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, 1, 2, s]} = 2\,\frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

$$\text{Bound[C, 1, -2, s]} = 2\,\frac{\text{Bound[WeightedOpenTriangle, 1, s]}}{2\,d\,\text{gjz[s]}};$$

```
, {s, {i, o}}]
```

Initial Blocks

Here we define the initial and terminal part of the diagrams as described in Appendix C.3.

```
In[4229]:= Do[
  Bound[Start, 0, s] = (1 + Bound[Double, 2, s]);
  Bound[Start, -1, s] = Bound[Loop, 4, s] + Bound[Bubble, 3, s];
  Bound[Start, -2, s] = Bound[Double, 4, s] + Bound[Triangle, 4, s];

  Do[
   Bound[P1, a, s] = Sum[Bound[Start, c, s] × Bound[A, c, a, s], {c, {-2, -1, 0}}];
   , {a, {-2, -1, 0, 1, 2}}];
  , {s, {i, o}}];
```

For the weighted diagram we use the same cases. For $\underline{b} \neq 0$ we have to split the weight on 0⟷x into 0⟷$\underline{b}$ and $\underline{b}$⟷x, which produces an extra factor 2.

```
In[4230]:= Do[Do[
     aprime = -Abs[a];
     Bound[DeltaStart, a, s] =
      Bound[C, 0, a, s] +
       If[aprime == 0, Bound[WeightedDouble, 1, s], 2 Bound[WeightedBubble, 1, s]] ×
        Bound[A, aprime, 0, s] +
       If[aprime == 0, 1, 2] × (Bound[Loop, 4, s] + Bound[Bubble, 4, s]/2) ×
        Bound[C, aprime, 0, s] (*b=v≠0*) +
       2 Bound[Loop, 4, s] × Bound[A, aprime, -1, s] +
       2 Bound[Loop, 4, s] × Bound[C, aprime, -1, s] (*dA(v,b)=1,v=0 *) +
       2 Bound[WeightedBubble, 2, s]/(2 d gjz[s]) Bound[A, aprime, -1, s] +
       2 Bound[Bubble, 3, s] × Bound[C, aprime, -1, s] (*dA(v,b)=1,v≠0 *) +
       2 Bound[WeightedBubble, 2, s] × Bound[A, aprime, -2, s] +
       2 Bound[Bubble, 4, s]/2 Bound[C, aprime, -2, s] (*dA(v,b)≥ 2,v=0*) +
       2 Bound[WeightedOpenBubble, 1, s] × Bound[A, aprime, -2, s] +
       2 Bound[Triangle, 4, s] × Bound[C, aprime, -2, s] (*dA(v,b)≥ 2,v≠0*);
     , {a, {-2, -1, 0, 1, 2}}];
   , {s, {i, o}}];
```

Initial Iota Block without weight

Here we define the bound of the initial part of the coefficients $\Xi^{(N),\iota}$ and $\Pi^{(N),\iota,\kappa}$, as given in Appendix C.4 of (II).

```
In[4231]:= Do[
   Do[
    Bound[B, 1, a, s] = Bound[tG, {1}, 1, s] × Bound[A, 0, a, s];
    , {a, {-2, -1, 0, 1, 2}}];
   Bound[B, 2, 0, s] = 1/(2 d) Bound[tG, {1}, 1, s]/gjz[s]
     (Bound[Bubble, 3, s] + Bound[Triangle, 3, s]);
   Bound[B, 2, -1, s] = Bound[tG, {1}, 1, s]/gjz[s] Bound[Triangle, 3, s]/(2 d);
   Bound[B, 2, -2, s] = Bound[tG, {1}, 1, s]/gjz[s] Bound[Square, 4, s]/(2 d);
   Bound[B, 2, 1, s] = Bound[B, 2, -1, s];
   Bound[B, 2, 2, s] = Bound[B, 2, -2, s];
   Bound[B, 3, 0, s] =
    Bound[Bubble, 3, s]/(2 d gjz[s]) (Bound[OpenBubble, 2, s] + Bound[OpenTriangle, 2, s]);
   Bound[B, 3, -1, s] = Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenTriangle, 2, s];
   Bound[B, 3, -2, s] = Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenSquare, 3, s];
   Bound[B, 3, 1, s] = Bound[B, 3, -1, s];
```

```
Bound[B, 3, 2, s] = Bound[B, 3, -2, s];
Bound[B, 4, 0, s] =
  1/(2 d) (Bound[Loop, 4, s] + 2 Bound[Bubble, 3, s] + Bound[Triangle, 3, s]);
Bound[B, 4, 1, s] =
  1/(2 d) (Bound[Loop, 4, s] + 2 Bound[Bubble, 3, s] + Bound[Triangle, 4, s]);
Bound[B, 4, 2, s] =
  1/(2 d) (Bound[Bubble, 3, s] + 2 Bound[Triangle, 4, s] + Bound[Square, 5, s]);
Bound[B, 4, -1, s] =
  1/(2 d) (Bound[Loop, 4, s] + Bound[Bubble, 4, s] + Bound[Triangle, 3, s]);
Bound[B, 4, -2, s] = 1/(2 d) (2 Bound[Triangle, 4, s] + Bound[Square, 5, s]);

SausageTobb = Bound[Double, 4, s];
(* a very crude bound on Σ_ικ B_2,2(e_ι+e_κ)*)
SausageWithPointTobb = Bound[Bubble, 4, s]/(2 d) Bound[tG, max, 1, s];
(* a double connection, with a line to e_ι*)

Bound[B, 5, 0, s] =
  gjz[s] (SausageWithPointTobb Bound[OpenTriangle, 0, s] +
     SausageTobb (Bound[Bubble, 3, s]/(2 d gjz[s]) + Bound[Triangle, 2, s]/(2 d gjz[s]) +
        Bound[Triangle, 1, s]/(2 d gjz[s])));
Bound[B, 5, 1, s] =
  gjz[s] (SausageWithPointTobb Bound[OpenTriangle, 1, s] +
     2 SausageTobb Bound[Triangle, 3, s]/(2 d gjz[s]));
Bound[B, 5, -1, s] = Bound[B, 5, 1, s];
Bound[B, 5, 2, s] =
  gjz[s] (SausageWithPointTobb Bound[OpenSquare, 2, s] +
     2 SausageTobb Bound[Square, 3, s]/(2 d gjz[s])) ;
Bound[B, 5, -2, s] = Bound[B, 5, 2, s];

Clear[SausageTobb, SausageWithPointTobb];

Do[
 Bound[B, 6, a, s] = Bound[tG, {1}, 1, s] × (Bound[P1, a, s] - Bound[A, 0, a, s])
```

```
(*u=0≠b*);
Bound[B, 7, a, s] =
 (Bound[Loop, 4, s]/(2 d) (*v=e_1*) +
    (2 d - 1)/(2 d) Bound[Loop, 4, s] (*one step to bb *) Bound[tG, max, 2, s] +
    Bound[Double, 4, s] × Bound[tG, max, 1, s] (*more steps to bb*))
  Bound[A, 0, a, s];
Bound[B, 8, a, s] =
 Bound[tG, {1}, 1, s] × Bound[tG, {1}, 3, s] (*bb=e1, v=0 *) Bound[A, -1, a, s] +
  Bound[Loop, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) (*bb=e1, dA(bb,v)=1, v≠0*)
   Bound[A, -1, a, s] + Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) (*bb=e1,
   dA(bb,v)>1, v≠0*) Bound[A, -2, a, s]
  + Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 1, s] (*bb≠e1*) Bound[A, -2, a, s];
Bound[B, 9, a, s] =
 (Bound[Loop, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, -1, a, s] (*v=e1,
    d(e1,bb)=1*) + Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, -2, a, s]
    (*v=e1, d(e1,bb)>1*)) +
  (*v=u≠ei*) Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 2, s] × Bound[A, -2, a, s];
Bound[B, 10, a, s] = Bound[Bubble, 3, s]/(2 d gjz[s])
  (Bound[tG, max, 2, s] × Bound[A, -1, a, s] +
    Bound[OpenBubble, 3, s] × Bound[A, -2, a, s]);
Bound[B, 11, a, s] = Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, -1, a, s]
  (*u=e1,d(v,bb)=1*) + Bound[Triangle, 5, s] Bound[tG, {1}, 1, s]/(2 d gjz[s])
   Bound[A, -2, a, s] (*u=e1,d(v,bb)>1*) +
  Bound[Bubble, 3, s]/(2 d gjz[s])
   (Bound[OpenBubble, 3, s] × Bound[A, -1, a, s] (*u≠e1,d(v,bb)=1*) +
     Bound[OpenTriangle, 4, s] × Bound[A, -2, a, s] (*u≠e1,d(v,bb)>1*));
Bound[B, 12, a, s] = Bound[B, 11, 0, s];
Bound[B, 13, a, s] =
 (((Bound[Bubble, 3, s]/(2 d gjz[s]))^2 (*u=e1*) +
    Bound[OpenBubble, 2, s] Bound[Triangle, 4, s]/(2 d gjz[s]) (*u≠e1*))
```

```
    Bound[A, -2, a, s];
  , {a, {-2, -1, 0, 1, 2}}];

Bound[B, 14, 0, s] =
  (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s] +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[tG, max, 2, s]) Bound[OpenTriangle, 1, s];
Bound[B, 14, -1, s] =
  (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s] +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[tG, max, 2, s]) Bound[OpenTriangle, 2, s];
Bound[B, 14, -2, s] =
  (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s] +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[tG, max, 2, s]) Bound[OpenSquare, 3, s];
Bound[B, 14, 1, s] = Bound[B, 14, -1, s];
Bound[B, 14, 2, s] = Bound[B, 14, -2, s];

Bound[B, 15, 0, s] = Bound[Triangle, 3, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s]
   (*u=e1, d(0,bb)=1*) + Bound[Square, 4, s] Bound[tG, {1}, 1, s] / (2 d gjz[s])
    Bound[tG, max, 2, s] (*u=e1, d(0,bb)>1*) +
   Bound[Double, 2, s] Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenTriangle, 1, s] (*u≠e1*);
Bound[B, 15, -1, s] =
  Bound[Triangle, 4, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s] +
   Bound[Square, 5, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[tG, max, 2, s] +
   Bound[Double, 2, s] Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenTriangle, 2, s];
Bound[B, 15, -2, s] = Bound[Square, 5, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[tG, {1}, 3, s]
   (*u=e1, d(0,bb)=1*) + Bound[OpenSquare, 4, s] Bound[tG, {1}, 1, s] / (2 d gjz[s])
    Bound[Double, 4, s] (*u=e1, d(0,bb)>1*) +
   Bound[Double, 2, s] Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenSquare, 3, s];
Bound[B, 15, 1, s] = Bound[B, 15, -1, s];
Bound[B, 15, 2, s] = Bound[B, 15, -2, s];
```

```
Bound[B, 16, 0, s] =
 Bound[Bubble, 3, s]
 ─────────────────── Bound[OpenBubble, 2, s] × Bound[OpenTriangle, 1, s] +
     2 d gjz[s]
  Bound[Triangle, 3, s] × Bound[OpenBubble, 2, s] × Bound[OpenTriangle, 1, s];
Bound[B, 16, -1, s] =
 Bound[Bubble, 3, s]
 ─────────────────── Bound[OpenBubble, 2, s] × Bound[OpenTriangle, 2, s] +
     2 d gjz[s]
  Bound[Triangle, 3, s] × Bound[OpenBubble, 2, s] × Bound[OpenTriangle, 2, s];
Bound[B, 16, -2, s] =
 Bound[Bubble, 3, s]
 ─────────────────── Bound[OpenBubble, 2, s] × Bound[OpenSquare, 3, s] +
     2 d gjz[s]
  Bound[Triangle, 3, s] × Bound[OpenBubble, 2, s] × Bound[OpenSquare, 3, s];
Bound[B, 16, 1, s] = Bound[B, 16, -1, s];
Bound[B, 16, 2, s] = Bound[B, 16, -2, s];

Do[
 Bound[P1Iota, a, s] = Sum[Bound[B, c, a, s], {c, 1, 16}];
 , {a, {-2, -1, 0, 1, 2}}];
, {s, {i, o}}];
```

In[4232]:=

```
Labeled[
 Grid[{Join[{"Part"}, Table[t, {t, 1, 16}]],
   Join[{"Abs."}, Table[NumberForm[Bound[B, t, 1, o], 3], {t, 1, 16}]],
   Join[{"% of Total"}, Table[NumberForm[100 * Bound[B, t, 2, o] / Bound[P1Iota, 2, o], 3],
     {t, 1, 16}]]}, Alignment → {Center}, Frame → True,
  Dividers → {{2 → True, -1 → True}, {2 → True}}, ItemStyle → {1 → Bold, 1 → Bold},
  Background → {{None}, {GrayLevel[0.9]}, {None}}],
 Style["Contribution to P1 ", Bold], Top] // Text
```

Out[4232]=

Contribution to P1

art	1	2	3	4	5	6	7	8	9	10	11	12	13
bs.	0.00-00-55-3	0.00-00-32-3	1.8×10^{-6}	0.00-00-52	1.61×10^{-8}	1.22×10^{-7}	3.36×10^{-8}	7.47×10^{-8}	3.16×10^{-8}	2.74×10^{-9}	3.6×10^{-8}	5.37×10^{-8}	3.88×10^{-9}
of To-tal	39.5	22.2	1.03	37.1	0.01-94	0.07-89	0.024	0.04-51	0.01-93	0.00-17-3	0.02-24	0.027	0.00-22-1

Initial weighted Iota Block

Here we implement the bounds given in Appendix C.4.2 of (II). We begin with the initial pieces in which $b = 0$

In[4233]:=

```
Do[Do[
  Bound[D, 1, 0, a, s] = Bound[tG, {1}, 1, s] × Bound[C, 0, a, s];
  Bound[D, 1, ei, a, s] =
```

```
  2 Bound[tG, {1}, 1, s] (Bound[C, 0, a, s] + gj[s] × Bound[A, a, 0, s]);
 Bound[D, 2, 0, a, s] = 1/(2 d) Bound[C, 0, a, s];
 Bound[D, 3, 0, a, s] = Bound[tG, {1}, 3, s] × Bound[C, -1, a, s] +
   Bound[Bubble, 4, s]/(2 d gjz[s]) Bound[C, -2, a, s];
 Bound[D, 3, ei, a, s] =
  2 Bound[D, 3, 0, a, s] +
   2 gj[s] × (Bound[tG, max, 2, s] × Bound[A, a, -1, s] +
      2 Bound[tG, max, 1, s] × Bound[A, a, -2, s]);
 Bound[D, 4, 0, a, s] = 1/(2 d) Bound[C, 0, a, s];
 , {a, {-2, -1, 0, 1, 2}}];
Bound[D, 1, ei, 0, s] =
 Bound[tG, {1}, 1, s] ( Bound[C, 0, a, s] + Bound[C, a, 0, s]);
Bound[D, 2, 0, 0, s] = (gjz[s] + Bound[tG, {1}, 1, s]) × Bound[tG, {1}, 3, s]
   (*x=e1*) + 1/(2 d) Bound[tG, {1}, 1, s]/gjz[s]
    (2 Bound[WeightedBubble, 2, s] (*w=e1≠x or w=x≠e1*) +
      Bound[WeightedTriangle, 3, s] ) (*e1≠w≠x or w=x≠e1*);
(*For M=1 && a=0 we can use symmetry to remove the factor 2,
created when splitting the weight ||x-eι||_2^2*)
Bound[D, 1, ei, 0, s] = Bound[tG, {1}, 1, s] × Bound[C, 0, 0, s] +
  Bound[tG, {1}, 1, s] × Bound[A, 0, 0, s];
Bound[D, 1, ei, -1, s] = 2 Bound[tG, {1}, 1, s] × Bound[C, 0, -1, s] +
  2 Bound[tG, {1}, 1, s]
   (Bound[Loop, 4, s]/(2 d gjz[s]) + Bound[Bubble, 4, s]/(2 d gjz[s]) + Bound[Triangle, 3, s]/(2 d gjz[s]));
Bound[D, 1, ei, -2, s] = 2 Bound[tG, {1}, 1, s] × Bound[C, 0, -2, s] +
  2 Bound[tG, {1}, 1, s] × (Bound[OpenBubble, 2, s] + Bound[OpenTriangle, 2, s]);
Bound[D, 1, ei, 1, s] = Bound[D, 1, ei, -1, s];
Bound[D, 1, ei, 2, s] = Bound[D, 1, ei, -2, s];

Bound[D, 2, ei, 0, s] = 1/(2 d) Bound[WeightedTriangle, 0, s];
Bound[D, 2, ei, -1, s] = 1/(2 d) Bound[WeightedTriangle, 1, s]/(2 d gjz[s]);
Bound[D, 2, ei, -2, s] = 1/(2 d) Bound[WeightedOpenTriangle, 0, s];
Bound[D, 2, ei, 1, s] = Bound[D, 2, ei, -1, s];
Bound[D, 2, ei, 2, s] = Bound[D, 2, ei, -2, s];

Bound[D, 4, ei, 0, s] = 1/(2 d) Bound[WeightedTriangle, 1, s];
Bound[D, 4, ei, -1, s] = 1/(2 d) Bound[WeightedTriangle, 2, s]/(2 d gjz[s]);
```

```
Bound[D, 4, ei, -2, s] = 1/(2 d) Bound[WeightedOpenTriangle, 1, s];
Bound[D, 4, ei, 1, s] = Bound[D, 4, ei, -1, s];
Bound[D, 4, ei, 2, s] = Bound[D, 4, ei, -2, s];
, {s, {i, o}}];
```

Then, we define the diagram for $b\neq 0$. These are trivial for lattice trees.

```
In[4234]:= Do[
SausageTobb = Bound[Double, 4, s];
(* a very crude bound on Σ_ικ B_2,2(e_ι+e_κ)*)
SausageWithPointTobb = Bound[Bubble, 4, s]/(2 d) Bound[tG, max, 1, s];
(* a double connection, with a line to e_ι*)


Bound[D, 5, ei, 0, s] =
 (2 SausageTobb Bound[WeightedTriangle, 1, s]/(2 d) (*v=b and v=0*) +
   Bound[WeightedOpenTriangle, 0, s] × gjz[s] SausageWithPointTobb);
Bound[D, 5, 0, 0, s] =
 (SausageTobb Bound[WeightedTriangle, 0, s]/(2 d) (*v=0*) +
   2 SausageTobb Bound[WeightedTriangle, 1, s]/(2 d) (*v=b, weight one *) +
   2 SausageTobb (Bound[Triangle, 2, s] + Bound[Bubble, 3, s])/(2 d) +
   Bound[WeightedOpenTriangle, 0, s]/(2 d) SausageWithPointTobb);
Bound[D, 5, ei, -1, s] =
 (2 SausageTobb Bound[WeightedTriangle, 2, s]/(2 d (2 d gjz[s])) (*v=b and v=0*) +
   Bound[WeightedOpenTriangle, 1, s]/(2 d) SausageWithPointTobb);
Bound[D, 5, 0, -1, s] =
 (SausageTobb Bound[WeightedTriangle, 1, s]/(2 d (2 d gjz[s])) (*v=0*) +
   2 SausageTobb Bound[WeightedTriangle, 2, s]/(2 d (2 d gjz[s])) (*v=b, weight one *) +
   2 SausageTobb (Bound[Triangle, 3, s] + Bound[Bubble, 4, s])/(2 d (2 d gjz[s])) +
```

```
    Bound[WeightedOpenTriangle, 1, s]
    --------------------------------- SausageWithPointTobb);
         2 d (2 d gjz[s])

Bound[D, 5, ei, -2, s] =
  (gjz[s] SausageTobb Bound[WeightedOpenTriangle, 0, s] +
    SausageTobb Bound[WeightedOpenTriangle, 1, s]/(2 d) (*v=0*) +
    gjz[s] SausageWithPointTobb Bound[WeightedOpenTriangle, 0, s]);
Bound[D, 5, 0, -2, s] =
  (2 SausageTobb Bound[WeightedOpenTriangle, 0, s]/(2 d) (*v=0 and v=b*) +
    SausageWithPointTobb Bound[WeightedOpenTriangle, 0, s]/(2 d)
     (1 + Bound[tG, max, 1, s]));

Bound[D, 5, ei, 1, s] = Bound[D, 5, ei, -1, s];
Bound[D, 5, ei, 2, s] = Bound[D, 5, ei, -2, s];
Bound[D, 5, 0, 1, s] = Bound[D, 5, 0, -1, s];
Bound[D, 5, 0, 2, s] = Bound[D, 5, 0, -2, s];
Clear[SausageTobb, SausageWithPointTobb];

Do[
 aprime = -Abs[a];
 Bound[D, 6, 0, a, s] =
  Bound[tG, {1}, 1, s] × (Bound[DeltaStart, a, s] - Bound[C, 0, a, s]);
 Bound[D, 6, ei, a, s] = 3/2 Bound[D, 6, 0, a, s] +
   3 Bound[tG, {1}, 1, s] ×
    (Bound[Double, 2, s] × Bound[A, a, 0, s] +
      (Bound[tG, {1}, 3, s] + Bound[Bubble, 3, s]/(2 d gjz[s])) × Bound[A, a, -1, s] +
      (Bound[tG, max, 2, s] + Bound[OpenBubble, 2, s]) × Bound[A, a, -2, s]) ;
 Bound[D, 7, 0, a, s] =
  2 (Bound[Loop, 4, s]/(2 d) (*v=e_1*) + Bound[Loop, 4, s] (*one step*)
      Bound[tG, max, 2, s] + Bound[Double, 4, s] × Bound[tG, max, 1, s]
     (*more steps*)) Bound[C, 0, a, s] +
   2 (Bound[Loop, 4, s]/(2 d) (*v=e_1*) +
```

```
      Bound[WeightedBubble, 2, s]
      --------------------------- Bound[tG, max, 2, s] (*more steps*))
             2 d gjz[s]
    Bound[A, a, 0, s];

Bound[D, 7, ei, a, s] =
  (Bound[Loop, 4, s]
   ----------------- (*v=e_1*) + 2 Bound[Loop, 4, s] (*one step*)
         2 d
     Bound[tG, max, 2, s] + 2 Bound[Double, 4, s] × Bound[tG, max, 1, s]
     (*more steps*)) Bound[C, 0, a, s] +
  2 (2 d - 1
     ------- Bound[Loop, 4, s] (*one step*) Bound[tG, max, 2, s] +
       2 d
      Bound[WeightedBubble, 3, s]
      --------------------------- Bound[tG, max, 2, s] (*more steps*))
             2 d gjz[s]
    Bound[A, a, 0, s];
                                               Bound[tG, {1}, 1, s]
Bound[D, 8, ei, a, s] = Bound[Loop, 4, s] -------------------- Bound[C, -1, a, s]
                                                2 d gjz[s]
                                                  Bound[tG, {1}, 1, s]
   (*bb=e_1, d(u,v)=1*) + Bound[Bubble, 3, s] --------------------
                                                     2 d gjz[s]
   (*bb=e_1,d(u,v)>1 *) Bound[C, -2, a, s] +
  2
   (Bound[Bubble, 3, s]
    ------------------- (Bound[OpenBubble, 1, s] × Bound[C, -1, a, s] +
        2 d gjz[s]
       Bound[OpenBubble, 2, s] × Bound[C, -2, a, s]) (* weight right *) +
     (Bound[Loop, 4, s]
      ----------------- Bound[A, a, -1, s] +
         2 d gjz[s]
       Bound[WeightedBubble, 3, s]
       --------------------------- Bound[A, a, -2, s])
              2 d gjz[s]
      (* weigth left v==0*) +
     Bound[WeightedBubble, 3, s]
     --------------------------- Bound[tG, max, 1, s] ×
            2 d gjz[s]
      (Bound[A, a, -1, s] + Bound[A, a, -2, s]) (*v≠0*)) (*bb≠e_1*);

Bound[D, 8, 0, a, s] =
 2 Bound[tG, {1}, 1, s] × Bound[tG, {1}, 3, s] × Bound[C, -1, a, s] +
                            Bound[tG, {1}, 1, s]
  2 Bound[tG, {1}, 3, s] -------------------- Bound[A, a, -1, s] (*bb=e_1,
                                  gjz[s]
  v=0 *) +
                         Bound[tG, {1}, 1, s]
  2 (Bound[Loop, 4, s] -------------------- Bound[C, -2, a, s] +
                              2 d gjz[s]
                           Bound[tG, {1}, 1, s]
     Bound[Bubble, 4, s] -------------------- Bound[C, -2, a, s] +
                                2 d gjz[s]
                         Bound[tG, {1}, 1, s]
     Bound[Loop, 4, s] -------------------- Bound[A, -1, a, s] +
                              2 d gjz[s]
```

```
      Bound[tG, max, 2, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, -2, a, s])
    (*bb=e1, v≠0*) +
   2
    (Bound[Bubble, 3, s]/(2 d gjz[s]) (Bound[tG, max, 1, s] × Bound[C, -1, a, s] +
        Bound[OpenBubble, 2, s] × Bound[C, -2, a, s]) (* weight right *) +
      (Bound[Loop, 4, s]/(2 d gjz[s]) Bound[A, a, -1, s] +
        Bound[WeightedBubble, 3, s]/(2 d gjz[s]) Bound[A, a, -2, s])
      (* weigth left v==0*) +
      Bound[WeightedBubble, 3, s]/(2 d gjz[s]) Bound[tG, max, 1, s] ×
        (Bound[A, a, -1, s] + Bound[A, a, -2, s]) (*v≠0*)) (*bb≠e1*);
Bound[D, 9, 0, a, s] =
  2 (Bound[Loop, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -1, a, s] +
      Bound[tG, {1}, 1, s]
       (2 × (2 d - 2) Bound[tG, {2}, 2, s] + 4 Bound[tG, {0, 1}, 4, s])
       Bound[A, a, -1, s]/(2 d)) (*v=u=ei,d(bb,v)=1*) +
   2 Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -2, a, s] +
   2 (Bound[WeightedBubble, 3, s] Bound[tG, {1}, 1, s]/(2 d gjz[s])) (*v=u=ei,
    d(bb,v)>1*) Bound[Abar, -2, a, s] +
   2 Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 2, s] × Bound[C, -2, a, s] +
   2 Bound[WeightedBubble, 2, s]/(2 d gjz[s]) Bound[tG, max, 1, s] × Bound[A, a, -1, s] +
   2 Bound[WeightedOpenBubble, 1, s] × Bound[tG, max, 1, s] × Bound[A, a, -2, s];
Bound[D, 9, ei, a, s] =
  2 (Bound[Loop, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -1, a, s] +
      Bound[Loop, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, a, -1, s]/(2 d gjz[s]))
    (*v=u=ei,d(bb,v)=1*) +
   2 Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -2, a, s] +
   2 (Bound[WeightedBubble, 2, s] Bound[tG, {1}, 1, s]/(2 d gjz[s])) (*v=u=ei,
    d(bb,v)>1*) Bound[Abar, -2, a, s] +
```

```
   3 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[OpenBubble, 2, s] × Bound[C, -2, a, s] +
   3 ((Bound[WeightedBubble, 2, s])/(2 d gjz[s]) Bound[tG, max, 1, s] × Bound[A, a, -1, s] +
      Bound[WeightedOpenBubble, 1, s] × Bound[tG, max, 1, s] ×
       Bound[A, a, -2, s]) + 3 (*v=u≠e_i*) (Bound[Bubble, 3, s])/(2 d gjz[s])
    Bound[tG, max, 1, s] × (Bound[A, a, -1, s] + Bound[A, a, -2, s]);
Bound[D, 10, 0, a, s] =
  2 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, max, 2, s] ×
    (Bound[C, -1, a, s] + (Bound[A, a, -1, s])/(2 d gjz[s])) (*d(bb,0)=1*) +
   2 (Bound[Bubble, 3, s])/(2 d gjz[s])
    (Bound[OpenBubble, 3, s] × Bound[C, -2, a, s] +
      Bound[WeightedOpenBubble, 1, s] × Bound[Abar, -2, a, s]);
Bound[D, 10, ei, a, s] =
  2 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, max, 2, s] ×
    (Bound[C, -1, a, s] + ((2 * (2 d - 2) + 4))/(2 d) (Bound[A, a, -1, s])/(2 d gjz[s])) +
   3 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[OpenBubble, 3, s] × Bound[C, -2, a, s] +
   3 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[WeightedOpenBubble, 2, s] ×
    Bound[Abar, -2, a, s] +
   3 (Bound[WeightedBubble, 3, s])/(2 d gjz[s]) Bound[OpenBubble, 3, s] ×
    Bound[Abar, -2, a, s];
Bound[D, 11, 0, a, s] =
  2 Bound[Bubble, 4, s] (Bound[tG, {1}, 1, s])/(2 d gjz[s]) Bound[C, -1, a, s] +
   2 Bound[WeightedBubble, 3, s] (Bound[tG, {1}, 1, s])/(2 d gjz[s]) (Bound[A, a, -1, s])/(2 d gjz[s])
   (*u=e_1,d(v,bb)=1*) + 2 Bound[Triangle, 5, s] (Bound[tG, {1}, 1, s])/(2 d gjz[s])
    Bound[C, -2, a, s] + 2 Bound[WeightedOpenBubble, 2, s] (Bound[tG, {1}, 1, s])/(2 d gjz[s])
    Bound[A, a, -2, s] (*u=e_1,d(v,bb)>1*) +
   2 ((Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[OpenBubble, 3, s] × Bound[C, -1, a, s] +
      (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[WeightedOpenBubble, 2, s] (Bound[A, a, -1, s])/(2 d gjz[s])
```

```
    (*u≠e1,d(v,bb)=1*) + Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenTriangle, 4, s] ×
     Bound[C, -2, a, s] + Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] ×
     Bound[A, a, -2, s] (*u≠e1,d(v,bb)>1*));
Bound[D, 11, ei, a, s] =
 2 Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -1, a, s] +
  2 Bound[WeightedBubble, 2, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, a, -1, s]/(2 d gjz[s]) +
  2 Bound[Triangle, 5, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -2, a, s] +
  2 Bound[WeightedOpenBubble, 1, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, a, -2, s] +
  3 (Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 3, s] × Bound[C, -1, a, s] +
     Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 2, s] Bound[A, a, -1, s]/(2 d gjz[s])
    (*u≠e1,d(v,bb)=1*) + Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenTriangle, 4, s] ×
     Bound[C, -2, a, s] + Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] ×
     Bound[A, a, -2, s] (*u≠e1,d(v,bb)>1*)) +
  3 Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenTriangle, 3, s] × Bound[Abar, a, -2, s];

Bound[D, 12, 0, a, s] =
 2 Bound[Bubble, 4, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -1, a, s] +
  2 Bound[WeightedBubble, 2, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, a, -1, s]
   (*u=e1,d(v,bb)=1*) +
  2 (Bound[Triangle, 5, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[C, -2, a, s] +
     Bound[WeightedOpenBubble, 1, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[A, a, -2, s])
    (*u=e1,d(v,bb)>1*) +
  3 (Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 3, s] × Bound[C, -1, a, s] +
     Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 2, s] Bound[A, a, -1, s]/(2 d gjz[s]) +
     Bound[WeightedBubble, 2, s]/(2 d gjz[s]) Bound[OpenBubble, 3, s] Bound[A, a, -1, s]/(2 d gjz[s]))
```

```
      (*u≠e1,d(v,bb)=1*) +
    3 (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenTriangle, 4, s] × Bound[C, -2, a, s] +
       Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] × Bound[A, a, -2, s] +
       Bound[WeightedBubble, 2, s] / (2 d gjz[s]) Bound[OpenTriangle, 4, s] ×
         Bound[Abar, a, -2, s]);
Bound[D, 12, ei, a, s] =
  2 Bound[Bubble, 4, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[C, -1, a, s] +
    2 Bound[WeightedBubble, 3, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[A, a, -1, s]
    (*u=e1,d(v,bb)=1*) +
    2 (Bound[Triangle, 5, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[C, -2, a, s] +
       Bound[WeightedOpenBubble, 2, s] Bound[tG, {1}, 1, s] / (2 d gjz[s]) Bound[A, a, -2, s])
      (*u=e1,d(v,bb)>1*) +
    3 (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenBubble, 3, s] × Bound[C, -1, a, s] +
       Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 2, s] Bound[A, a, -1, s] / (2 d gjz[s]) +
       Bound[WeightedBubble, 2, s] / (2 d gjz[s]) Bound[OpenBubble, 3, s] Bound[A, a, -1, s] / (2 d gjz[s]))
      (*u≠e1,d(v,bb)=1*) +
    3 (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenTriangle, 4, s] × Bound[C, -2, a, s] +
       Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] × Bound[A, a, -2, s] +
       Bound[WeightedBubble, 2, s] / (2 d gjz[s]) Bound[OpenTriangle, 4, s] ×
         Bound[Abar, a, -2, s]);
Bound[D, 13, 0, a, s] = 2 (Bound[Bubble, 3, s] / (2 d gjz[s]))^2 Bound[C, -2, a, s] +
    2 Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedBubble, 2, s] / (2 d gjz[s]) Bound[Abar, a, -2, s]
    (*u=e1*) +
    2 Bound[OpenBubble, 2, s] Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[C, -2, a, s] +
```

```
  2 Bound[WeightedOpenBubble, 1, s] (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[Abar, a, -2, s]
  (*u≠e1*);
 Bound[D, 13, ei, a, s] = 2 ((Bound[Bubble, 3, s])/(2 d gjz[s]))^2 Bound[C, -2, a, s] +
  2 (Bound[Bubble, 3, s])/(2 d gjz[s]) (Bound[WeightedBubble, 2, s])/(2 d gjz[s]) Bound[Abar, a, -2, s]
  (*u=e1*) +
  3 Bound[OpenBubble, 2, s] (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[C, -2, a, s] +
  3 Bound[WeightedOpenBubble, 1, s] (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[Abar, a, -2, s]
  (*u≠e1*) + 3 Bound[OpenBubble, 2, s] (Bound[Triangle, 4, s])/(2 d gjz[s])
   Bound[Abar, -2, a, s] (*u≠e1*);
 Bound[D, 14, 0, a, s] =
  2 ((Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, {1}, 3, s] +
     (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[tG, max, 2, s])
   Bound[WeightedOpenTriangle, 0, s] +
  2 ((Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, {1}, 3, s] +
     Bound[OpenBubble, 2, s] × Bound[WeightedBubble, 2, s])
   Bound[Abar, a, -2, s];
 Bound[D, 14, ei, a, s] =
  (2 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, {1}, 3, s] +
     3 (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[tG, max, 2, s])
   Bound[WeightedOpenTriangle, 0, s] +
  2 Bound[OpenBubble, 2, s] × Bound[WeightedBubble, 2, s] ×
   Bound[Abar, a, -2, s] +
  (2 (Bound[Bubble, 3, s])/(2 d gjz[s]) Bound[tG, {1}, 3, s] (2 × (2 d - 2) + 4)/(2 d) +
     3 (Bound[Triangle, 4, s])/(2 d gjz[s]) Bound[tG, max, 2, s]) Bound[Abar, a, -2, s];

 , {a, {-2, -1, 0, 1, 2}}];

Bound[D, 15, 0, 0, s] =
```

```
 2 Bound[WeightedOpenTriangle, 1, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[Double, 2, s] +
  2 Bound[OpenTriangle, 2, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[WeightedDouble, 1, s]
  (*u=e1*) + 2 Bound[WeightedOpenTriangle, 0, s] × Bound[Double, 2, s]
  Bound[Bubble, 3, s]/(2 d gjz[s]) (*u≠e1*) +
  2 Bound[OpenTriangle, 1, s] Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedDouble, 1, s]
  (*u≠e1*);
(*For a=1,-1*)
Bound[D, 15, 0, -1, s] =
 2 Bound[WeightedOpenTriangle, 2, s]/(2 d gjz[s]) Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[Double, 2, s] +
  2 Bound[OpenTriangle, 3, s]/(2 d gjz[s]) Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[WeightedDouble, 1, s]
  (*u=e1*) + 2 Bound[WeightedOpenTriangle, 1, s]/(2 d gjz[s]) Bound[Double, 2, s]
  Bound[Bubble, 3, s]/(2 d gjz[s]) (*u≠e1*) +
  2 Bound[OpenTriangle, 2, s]/(2 d gjz[s]) Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedDouble, 1, s]
  (*u≠e1*);

(*For a=2,-2*)
Bound[D, 15, 0, -2, s] =
 2 Bound[WeightedOpenTriangle, 1, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[Double, 2, s] +
  2 (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 2, {0}] × Bound[tG, {1}, 1, s] ×
   Bound[WeightedDouble, 1, s] (*u=e1*) +
  2 Bound[WeightedOpenTriangle, 0, s] × Bound[Double, 2, s] Bound[Bubble, 3, s]/(2 d gjz[s])
  (*u≠e1*) + 2 (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 1, {0}] Bound[Bubble, 3, s]/(2 d gjz[s])
   Bound[WeightedDouble, 1, s] (*u≠e1*);
Bound[D, 15, ei, 0, s] = 3/2 Bound[D, 15, 0, 0, s] + 3 Bound[B, 15, 0, s];
Bound[D, 15, ei, -1, s] = 3/2 Bound[D, 15, 0, -1, s] + 3 Bound[B, 15, -1, s]/(2 d gjz[s]);

Bound[D, 15, ei, -2, s] =
 3/2 Bound[D, 15, 0, -2, s] +
  3 (Bound[OpenTriangle, 3, s] Bound[tG, {1}, 1, s]/(2 d gjz[s]) Bound[tG, {1}, 3, s] +
```

```
(2 d gjz[s])^2 VarGamma2[s]^3 K[3, 2, {0}] Bound[tG, {1}, 1, s] / (2 d gjz[s])
 Bound[Double, 4, s] + Bound[Double, 2, s] Bound[Bubble, 3, s] / (2 d gjz[s])
 (2 d gjz[s])^1 VarGamma2[s]^3 K[3, 1, {0}]);
Bound[D, 15, 0, 1, s] = Bound[D, 15, 0, -1, s];
Bound[D, 15, 0, 2, s] = Bound[D, 15, 0, -2, s];
Bound[D, 15, ei, 1, s] = Bound[D, 15, ei, -1, s];
Bound[D, 15, ei, 2, s] = Bound[D, 15, ei, -2, s];

Bound[D, 16, 0, 0, s] =
 2
  (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenBubble, 2, s] ×
     Bound[WeightedOpenTriangle, 0, s] (*d(0,v)=1*) +
    Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] ×
     Bound[OpenTriangle, 1, s] (*d(0,v)=1*) +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[OpenBubble, 2, s] ×
     Bound[WeightedOpenTriangle, 0, s] (*d(0,v)>1*) +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] ×
     Bound[OpenTriangle, 1, s] (*d(0,v)>1*));

Bound[D, 16, 0, -1, s] =
 2
  (Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[OpenBubble, 2, s]
     Bound[WeightedOpenTriangle, 1, s] / (2 d gjz[s]) +
    Bound[Bubble, 3, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s]
     Bound[OpenTriangle, 2, s] / (2 d gjz[s]) +
    Bound[WeightedTriangle, 3, s] × Bound[OpenBubble, 2, s]
     Bound[OpenTriangle, 2, s] / (2 d gjz[s]) +
    Bound[Triangle, 4, s] / (2 d gjz[s]) Bound[WeightedOpenBubble, 1, s]
     Bound[WeightedOpenTriangle, 2, s] / (2 d gjz[s]));
```

```
Bound[D, 16, 0, -2, s] =
 2
  (Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 2, s] ×
     Bound[WeightedOpenTriangle, 0, s] +
    Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] (2 d gjz[s])^1
     VarGamma2[s]^3 K[3, 1, {0}] +
    Bound[Triangle, 3, s] × Bound[OpenBubble, 2, s] ×
     Bound[WeightedOpenTriangle, 0, s] +
    Bound[Triangle, 4, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 1, s] (2 d gjz[s])^1
     VarGamma2[s]^3 K[3, 1, {0}]);

Bound[D, 16, ei, 0, s] = 3/2 Bound[D, 16, 0, 0, s] + 3 Bound[B, 16, 0, s];
Bound[D, 16, ei, -1, s] = 3/2 Bound[D, 16, 0, -1, s] + 3 Bound[B, 16, -1, s]/(2 d gjz[s]);
Bound[D, 16, ei, -2, s] =
 3/2 Bound[D, 16, 0, -2, s] +
  3 (Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 2, s] +
     Bound[Triangle, 3, s] × Bound[OpenBubble, 2, s]) (2 d gjz[s])^1
   VarGamma2[s]^3 K[3, 1, {0}];


Bound[D, 16, 0, 1, s] = Bound[D, 16, 0, -1, s];
Bound[D, 16, 0, 2, s] = Bound[D, 16, 0, -2, s];
Bound[D, 16, ei, 1, s] = Bound[D, 16, ei, -1, s];
Bound[D, 16, ei, 2, s] = Bound[D, 16, ei, -2, s];

Do[
 Bound[D, 0, a, s] = Sum[Bound[D, c, 0, a, s], {c, 1, 16}];
 Bound[D, ei, a, s] = Sum[Bound[D, c, ei, a, s], {c, 1, 16}];
 , {a, {-2, -1, 0, 1, 2}}];
, {s, {i, o}}];
```

```
In[4235]:= Labeled[
  Grid[{Join[{"Part"}, Table[t, {t, 1, 8}]],
    Join[{"Abs."}, Table[NumberForm[Bound[D, t, 0, 2, o], 3], {t, 1, 8}]],
    Join[{"% of Total"}, Table[NumberForm[100 * Bound[D, t, 0, 2, o] / Bound[D, 0, 2, o], 3],
      {t, 1, 8}]]}, Alignment → {Center}, Frame → True,
   Dividers → {{2 → True, -1 → True}, {2 → True}}, ItemStyle → {1 → Bold, 1 → Bold},
   Background → {{None}, {GrayLevel[0.9]}, {None}}],
  Style["Contribution to D^(I,2) ", Bold], Top] // Text
Labeled[
  Grid[{Join[{"Part"}, Table[t, {t, 9, 16}]],
    Join[{"Abs."}, Table[NumberForm[Bound[D, t, 0, 2, o], 3], {t, 9, 16}]],
    Join[{"% of Total"}, Table[NumberForm[100 * Bound[D, t, 0, 2, o] / Bound[D, 0, 2, o], 3],
      {t, 9, 16}]]}, Alignment → {Center}, Frame → True,
   Dividers → {{2 → True, -1 → True}, {2 → True}}, ItemStyle → {1 → Bold, 1 → Bold},
   Background → {{None}, {GrayLevel[0.9]}, {None}}],
  Style["Contribution to D^(I,2) ", Bold], Top] // Text

Labeled[
  Grid[{Join[{"Part"}, Table[t, {t, 1, 8}]],
    Join[{"Abs."}, Table[NumberForm[Bound[D, t, ei, 2, o], 3], {t, 1, 8}]],
    Join[{"% of Total"}, Table[NumberForm[100 * Bound[D, t, ei, 2, o] / Bound[D, ei, 2, o], 3],
      {t, 1, 8}]]}, Alignment → {Center}, Frame → True,
   Dividers → {{2 → True, -1 → True}, {2 → True}}, ItemStyle → {1 → Bold, 1 → Bold},
   Background → {{None}, {GrayLevel[0.9]}, {None}}],
  Style["Contribution to D^(II,2) ", Bold], Top] // Text
Labeled[
  Grid[{Join[{"Part"}, Table[t, {t, 9, 16}]],
    Join[{"Abs."}, Table[NumberForm[Bound[D, t, ei, 2, o], 3], {t, 9, 16}]],
    Join[{"% of Total"}, Table[NumberForm[100 * Bound[D, t, ei, 2, o] / Bound[D, ei, 2, o], 3],
      {t, 9, 16}]]}, Alignment → {Center}, Frame → True,
   Dividers → {{2 → True, -1 → True}, {2 → True}}, ItemStyle → {1 → Bold, 1 → Bold},
   Background → {{None}, {GrayLevel[0.9]}, {None}}],
  Style["Contribution to D^(II,2) ", Bold], Top] // Text
```

Out[4235]=

Contribution to $D^{I,2}$

Part	1	2	3	4	5	6	7	8
Abs.	0.0124	0.0117	0.000344	0.0117	5.9×10^{-6}	0.0000501	0.0000152	0.0000543
% of Total	34.2	32.1	0.947	32.1	0.0162	0.138	0.0417	0.149

Out[4236]=

Contribution to $D^{I,2}$

Part	9	10	11	12	13	14	15	16
Abs.	0.000062-8	3.78×10^{-6}	0.000013-5	0.000017-3	8.84×10^{-6}	5.98×10^{-6}	7.64×10^{-6}	0.000010-6
% of Total	0.173	0.0104	0.0372	0.0476	0.0243	0.0165	0.021	0.0292

Out[4237]=

Contribution to $D^{II,2}$

Part	1	2	3	4	5	6	7	8
Abs.	0.0249	0.0117	0.00119	0.000278	3.15×10^{-6}	0.0000757	0.000011	0.0000425
% of Total	65.	30.4	3.11	0.725	0.0082	0.197	0.0285	0.111

Out[4238]=

Contribution to $D^{II,2}$

Part	9	10	11	12	13	14	15	16
Abs.	0.000078-6	6.14×10^{-6}	0.000015	0.000016-9	0.000012-1	7.42×10^{-6}	0.000012-3	0.000018-1
% of Total	0.205	0.016	0.0391	0.0439	0.0315	0.0193	0.0321	0.0473

Definition of the vectors and matrices

We condition on the length of the backbone and identify whether the backbone is on the top or bottom of the diagram.

index	interpretation
-2	backbone is on bottom, d (u, v) ≥ 2.
-1	backbone is on bottom, d (u, v) = 1.
0	u = v
1	backbone is on top, d (u, v) = 1.
2	backbone in on the top, d (u, v) ≥ 2.

In[4239]:=

```
Do[
 Matrix[A, s] = Table[Bound[A, a, b, s], {a, -2, 2}, {b, -2, 2}];
 Matrix[Abar, s] = Table[Bound[Abar, a, b, s], {a, -2, 2}, {b, -2, 2}];
 Matrix[C, s] = Table[Bound[C, a, b, s], {a, -2, 2}, {b, -2, 2}];

 Vector[P1, s] = Table[Bound[P1, a, s], {a, -2, 2}];
 Vector[DeltaStart, s] = Table[Bound[DeltaStart, a, s], {a, -2, 2}];
 Vector[EndOpen, s] = Table[Bound[A, b, 0, s], {b, -2, 2}];
 Vector[EndClosed, s] = Table[Bound[A, 0, b, s], {b, -2, 2}];
 Vector[DeltaEnd, s] = Table[Bound[C, b, 0, s], {b, -2, 2}];

 Vector[P1Iota, s] = Table[Bound[P1Iota, a, s], {a, -2, 2}];
 Vector[D, 0, s] = Table[Bound[D, 0, a, s], {a, -2, 2}];
 Vector[D, ei, s] = Table[Bound[D, ei, a, s], {a, -2, 2}];
 , {s, {i, o}}]
```

To compute the sum over matrices we compute a representation of the opening and closing vectors using eigenvalues of the matrices *A*. If the matrix is not invertible such a representation (using real values only) does not need to exist. We bypass this problem by using a symmetric matrix that dominates *A*, see *ASym* below, and use this in our bound for $N \geq 4$.

In[4240]:=
```
Do[
 Matrix[ASym, s] = Table[Max[Bound[A, a, b, s], Bound[A, b, a, s]],
   {a, -2, 2}, {b, -2, 2}];

 EigenA[s] = Eigensystem[Transpose[Matrix[ASym, s]]];
 InverseProduct[left, s] = Inverse[Transpose[EigenA[s][[2]]]].Vector[P1, s];
 InverseProduct[iota, s] = Inverse[Transpose[EigenA[s][[2]]]].Vector[P1Iota, s];
 InverseProduct[right, s] =
  Inverse[Transpose[EigenA[s][[2]]]].Vector[EndClosed, s];
 Do[
  EigenVector[left, j, s] = EigenA[s][[2, j]] * InverseProduct[left, s][[j]];
  EigenVector[iota, j, s] = EigenA[s][[2, j]] * InverseProduct[iota, s][[j]];
  EigenVector[right, j, s] = EigenA[s][[2, j]] * InverseProduct[right, s][[j]];
  EigenValue[j, s] = EigenA[s][[1, j]];
  , {j, 1, 5}]

 , {s, {i, o}}]
```

Bounds on the coefficients

Bounds for *N=0*

Here, we implement the bounds stated in Lemma 5.1 of (II).

In[4241]:=
```
Do[

 Bound[Xi, 0, s] = 1/gLower Bound[Double, 2, s];
 Bound[Xi, 0, Delta, s] = Bound[WeightedDouble, 1, s];
 Bound[Xi, R, 0, s] = 1/gLower Bound[Double, 4, s];
 Bound[Xi, R, 0, Delta, s] = Bound[WeightedDouble, 2, s];
 Bound[Psi, RII, 0, s] = Bound[Xi, R, 0, s];
 Bound[Psi, RII, 0, Delta, s] = Bound[Xi, R, 0, Delta, s];
 Bound[Psi, RI, 0, s] =
  muOverMub[s] ×
   (Bound[Loop, 4, s](*x=e1 does no contribute*) + Bound[Loop, 4, s]
     (*x=eκ+e1 one con. with two steps is excluded*) + Bound[Bubble, 6, s]/2);
 Bound[Psi, RI, 0, Delta, s] = Bound[Xi, 0, Delta, s] + Bound[Psi, RI, 0, s];
 , {s, {i, o}}]
```

Next, we implement the bounds on $\Xi^{(0),\iota}$ stated in Lemma 5.2 of (II):

In[4242]:=
```
Do[
  conAlpha[n_, m_] :=
```

```
  1/gLower ((*u=0*) Bound[tG, {1}, 1, s] × Bound[Double, 2 * n, s] +
     If[n == 0, Bound[Double, 2, s], 0] (*x=e_i*) + (*u=e_i≠x*)
      1/(2 d) Bound[tG, {1}, 1, s]/gjz[s] Bound[Bubble, 1 + n + Max[1, m], s] +
      1/gLower
       (If[n ≤ 1, Bound[tG, max, 2, s] (2 d - 1) gjz[s] × Bound[tG, {1}, 3, s], 0] +
         Bound[tG, max, 1, s] × Bound[Double, 4, s]) (*u=x≠{0,e_1}*) +
      Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, n + 1, s] (*u≠0,e_1,x and 0≠x*));

conBeta[n_] :=
  1/gLower (Bound[tG, {1}, 1, s] × Bound[WeightedDouble, n + 1, s] +
      If[n == 0, Bound[tG, {1}, 1, s] × Bound[tG, {1}, 3, s] (*u=e_1=x*), 0] +
      Bound[tG, {1}, 1, s] Bound[WeightedDouble, 2, s]/(2 d gjz[s]) (*u=e_1≠x*)) +
   1/gLower
    (If[n == 0, Bound[Loop, 4, s]/(2 d gjz[s]) Bound[tG, {1}, 3, s] (*u=x, d(0,x)=1*), 0] +
      Bound[WeightedBubble, 2, s]/(2 d gjz[s]) Bound[tG, max, 2, s] (*u=x, d(0,x)≥ 2*) +
      Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[WeightedOpenBubble, 1, s]);
conGamma[n_] :=
  (1/gLower Bound[tG, {1}, 1, s] × Bound[Double, 2, s] +
     Bound[tG, {1}, 1, s] × Bound[WeightedDouble, 1, s]) (*u=0*) +
   If[n == 1, Bound[tG, max, 1, s]
      (Bound[tG, {2}, 2, s]^2 (2 d - 2) + Bound[tG, {0, 1}, 4, s]^2) (*u=x*), 0] +
   Bound[tG, max, 2, s] Bound[WeightedBubble, 2, s]/(2 d gjz[s]) (*u=x*) +
   2 (Bound[WeightedOpenBubble, 1, s] Bound[Bubble, 3, s]/(2 d gjz[s]) +
      1/gLower Bound[Bubble, 3, s]/(2 d gjz[s]) Bound[OpenBubble, 2, s]) (*u≠e_i*);
Bound[XiIota, 0, s] = muOverMub[s] × Bound[tG, {1}, 1, s] + conAlpha[1, 0];
Bound[XiIota, RI, 0, s] = conAlpha[1, 2];
Bound[XiIota, RII, 0, s] = conAlpha[2, 1];
Bound[XiIota, 0, Delta, 0, s] = conBeta[0];
Bound[XiIota, 0, Delta, ei, s] = conGamma[1];
```

```
Bound[XiIota, RI, 0, Delta, ei, s] = conGamma[2];
Bound[XiIota, RII, 0, Delta, 0, s] = conBeta[1];
Bound[XiIota, alphaI, 0, Atei, s] = 1/gLower Bound[Double, 2, s];
Bound[XiIota, alphaII, 0, AtZero, s] = muOverMub[s] × Bound[tG, {1}, 1, s];

Bound[XiIota, alphaI, 0, SumAroundei, s] =
 muOverMub[s] × Bound[tG, {1}, 1, s] +
   1/gLower
   (Bound[Double, 4, s] × gjz[s] + 1/(2 d) Bound[tG, {1}, 1, s]/gjz[s] Bound[Loop, 4, s]);
Bound[XiIota, alphaII, 0, SumAroundZero, s] =
 1/gLower Bound[tG, {1}, 1, s] × Bound[tG, {1}, 3, s] +
   1/gLower 1/(2 d) Bound[tG, {1}, 1, s]/gjz[s] Bound[Loop, 4, s] +
   1/gLower Bound[tG, {1}, 1, s]/gjz[s] Bound[Bubble, 3, s] × Bound[tG, max, 1, s] +
   Bound[Loop, 4, s] × (Bound[tG, {1}, 1, s] + Bound[tG, {2}, 2, s]);
Bound[Pi, alpha, 0, s] = (2 d - 1) Bound[Double, 2, s] (*x=e1*) z[s];
Bound[Pi, R, 0, s] = (2 d - 1) gjz[s] × Bound[tG, {1}, 1, s] +
   gjz[s] × conAlpha[1, 1];
Bound[Pi, R, 0, Delta, eiek, s] =
 8 d^2 gjz[o] × Bound[tG, {1}, 1, s] + (2 d)^2 gjz[o] × conGamma[1] +
   4 d (d + 1) gjz[o] × conAlpha[1, 1];
, {s, {i, o}}];
```

Bounds for *N=1*

Implementation of the bounds stated in Lemma 5.3 of (II):

In[4243]:=
```
Do[
  Bound[Xi, 1, s] = muOverMub[s] × Bound[P1, 0, s];
  Bound[Xi, R, 1, s] =
   muOverMub[s] ×
    (Bound[P1, 0, s] - Bound[A, 0, 0, s] (*b≠0*) + 2 Bound[Bubble, 4, s]
      (*b=0,w in 0,x *) + Bound[Triangle, 4, s]);

  Bound[Psi, RI, 1, s] =
   muOverMub[s] ×
    (Bound[P1, 0, s] - Bound[A, 0, 0, s] (*b≠0*) + 4 Bound[Loop, 4, s] +
      2 Bound[Bubble, 5, s] + Bound[Bubble, 3, s] + Bound[Bubble, 4, s] +
      Bound[Triangle, 5, s]);
  Bound[Psi, RII, 1, s] =
   muOverMub[s] ×
    (Bound[P1, 0, s] - Bound[A, 0, 0, s] (*b≠0*) + 2 Bound[Bubble, 4, s]
      (*b=0,w in 0,x *) + Bound[Triangle, 4, s]);

  Bound[Xi, 1, Delta, s] = Bound[DeltaStart, 0, s];
  Bound[Xi, R, 1, Delta, s] = Bound[DeltaStart, 0, s];
  Bound[Psi, RI, 1, Delta, s] = Bound[Xi, 1, Delta, s] + Bound[Psi, RI, 1, s];
  Bound[Psi, RII, 1, Delta, s] = Bound[Xi, R, 1, Delta, s];

  Bound[XiIota, 1, s] = muOverMub[s] × Bound[P1Iota, 0, s];
  Bound[XiIota, 1, Delta, 0, s] = Bound[D, 0, 0, s];
  Bound[XiIota, 1, Delta, ei, s] = Bound[D, ei, 0, s];
  , {s, {i, o}}];
```

Bounds for $N = 2, 3$

We consider also *N=2,3* as a special case, as these are large enough to be of numerical significance. We first implement the bounds on the absolute value of the coefficient as stated in Proposition 5.4 of (II).

In[4244]:=
```
Do[
  Bound[Xi, 2, s] = muOverMub[s] × Vector[P1, s].Vector[EndOpen, s];
  Bound[Xi, 3, s] =
   muOverMub[s] × Vector[P1, s].Matrix[A, s].Vector[EndOpen, s];

  Bound[XiIota, 2, s] = muOverMub[s] × Vector[P1Iota, s].Vector[EndOpen, s];
  Bound[XiIota, 3, s] =
   muOverMub[s] × Vector[P1Iota, s].Matrix[A, s].Vector[EndOpen, s];
  , {s, {i, o}}];
```

We improve the bound on the weighted diagrams stated in Proposition 5.4 of (II) slightly. Thee bound stated in the proposition splits the weight $|x|_2^2$ using

$$|x|_2^2 \le 2|y|_2^2 + 2|x-y|_2^2$$

when splitting the weight along the different building block, see Figure 14 of (II). In the case that the shared lines collapses to a point (u=y, so that l=0) we can use

$$|x|_2^2 = |y|_2^2 + |x-y|_2^2 + 2\sum_i x_i y_i$$

By the spatial symmetry in all directions of the building blocks, that only exists in the full extent if the shared line is collapsed, the sum of $x_i\, y_i$ cancels out. Thus, the bounds stated in Proposition 5.4 of (II) is a factor 2 too big for this specific case. This improves the bound by around 10 percent. This is analogous to the improvement explained in (6.22) of (II).

In[4245]:=
```
Do[
  Bound[Xi, 2, Delta, s] =
   2 (Vector[P1, s].Vector[DeltaEnd, s] +
       Vector[DeltaStart, s].Vector[EndClosed, s]) -
    (Vector[P1, s][[3]] × Vector[DeltaEnd, s][[3]] +
      Vector[DeltaStart, s][[3]] × Vector[EndClosed, s][[3]]);
  Bound[Xi, 3, Delta, s] =
   3 (Vector[P1, s].Matrix[A, s].Vector[DeltaEnd, s] +
       Vector[P1, s].Matrix[C, s].Vector[EndClosed, s] +
       Vector[DeltaStart, s].Matrix[A, s].Vector[EndClosed, s]) -
    Sum[Vector[P1, s][[3]] × Matrix[A, s][[3, b]] × Vector[DeltaEnd, s][[b]] +
      Vector[P1, s][[3]] × Matrix[C, s][[3, b]] × Vector[EndClosed, s][[b]] +
      Vector[DeltaStart, s][[3]] × Matrix[A, s][[3, b]] × Vector[EndClosed, s][[b]]
     , {b, 1, 5}] -
    Sum[Vector[P1, s][[a]] × Matrix[A, s][[a, 3]] × Vector[DeltaEnd, s][[3]] +
      Vector[P1, s][[a]] × Matrix[C, s][[a, 3]] × Vector[EndClosed, s][[3]] +
      Vector[DeltaStart, s][[a]] × Matrix[A, s][[a, 3]] × Vector[EndClosed, s][[3]]
     , {a, 1, 5}];

  Do[
   Bound[XiIota, 2, Delta, type, s] =
    2 (Vector[P1Iota, s].Vector[DeltaEnd, s] +
        Vector[D, type, s].Vector[EndClosed, s]) -
     (Vector[P1Iota, s][[3]] × Vector[DeltaEnd, s][[3]] +
       Vector[D, type, s][[3]] × Vector[EndClosed, s][[3]]);
   Bound[XiIota, 3, Delta, type, s] =
    Bound[XiIota, 3, Delta, type, s] =
     3 (Vector[P1Iota, s].Matrix[A, s].Vector[DeltaEnd, s] +
         Vector[P1Iota, s].Matrix[C, s].Vector[EndClosed, s] +
         Vector[D, type, s].Matrix[A, s].Vector[EndClosed, s]) -
      Sum[Vector[P1Iota, s][[a]] × Matrix[A, s][[a, 3]] × Vector[DeltaEnd, s][[3]] +
        Vector[P1Iota, s][[a]] × Matrix[C, s][[a, 3]] × Vector[EndClosed, s][[3]] +
        Vector[D, type, s][[a]] × Matrix[A, s][[a, 3]] × Vector[EndClosed, s][[3]],
       {a, 1, 5}] -
      Sum[Vector[P1Iota, s][[3]] × Matrix[A, s][[3, b]] × Vector[DeltaEnd, s][[b]] +
        Vector[P1Iota, s][[3]] × Matrix[C, s][[3, b]] × Vector[EndClosed, s][[b]] +
        Vector[D, type, s][[3]] × Matrix[A, s][[3, b]] × Vector[EndClosed, s][[b]],
       {b, 1, 5}];
   , {type, {0, ei}}];
  , {s, {i, o}}];
```

Bounds for $N \geq 4$

Next, we bound the sum over all odd and even $N \geq 4$. We use the earlier computed decomposition of the vectors P and P^i in term of eigenvectors of A. We use these eigenvectors and the geometric sum to compute the sum of the bounds, see Section 5.4 of (I). We begin with the bound on the absolute value.

```
In[4246]:= Do[
  Do[
   vl[j] = Abs[EigenVector[left, j, s]];
   vi[j] = Abs[EigenVector[iota, j, s]];

   e[j] = Abs[EigenValue[j, s]];
   evi[j] = Abs[EigenValue[j, s]];
   , {j, 1, 5}];
  Bound[Xi, EvenTail, s] =
   muOverMub[s] × Sum[e[j]^2/(1 - e[j]^2) vl[j].Vector[EndOpen, s], {j, 1, 5}];
  Bound[Xi, OddTail, s] =
   muOverMub[s] × Sum[e[j]^3/(1 - e[j]^2) vl[j].Vector[EndOpen, s], {j, 1, 5}];
  Bound[XiIota, EvenTail, s] =
   muOverMub[s] × Sum[evi[j]^2/(1 - evi[j]^2) vi[j].Vector[EndOpen, s], {j, 1, 5}];
  Bound[XiIota, OddTail, s] =
   muOverMub[s] × Sum[evi[j]^3/(1 - evi[j]^2) vi[j].Vector[EndOpen, s], {j, 1, 5}];
  , {s, {i, o}}]
```

Then, we compute the bound on the weighted coefficients:

```
In[4247]:= Do[
  Do[
   vl[j] = Abs[EigenVector[left, j, s]];
   vr[j] = Abs[EigenVector[right, j, s]];
   vi[j] = Abs[EigenVector[iota, j, s]];
   e[j] = Abs[EigenValue[j, s]];
   evi[j] = Abs[EigenValue[j, s]];
   , {j, 1, 5}];
  Bound[Xi, EvenTail, Delta, s] =
   Vector[DeltaStart, s].Sum[vr[j] * e[j]^2 (2/(1 - e[j]^2)^2 + 2/(1 - e[j]^2)), {j, 1, 5}] +
    Sum[(2/(1 - e[j]^2)^2 + 2/(1 - e[j]^2)) * vl[j] e[j]^2, {j, 1, 5}].Vector[DeltaEnd, s] +
    Sum[2/(1 - e[j]^2)^2 * vl[j], {j, 1, 5}].
     (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
```

```
      Sum[2/(1 - e[j]^2) vr[j], {j, 1, 5}] +
    Sum[2/(1 - e[j]^2) * vl[j], {j, 1, 5}].
      (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
      Sum[2/(1 - e[j]^2)^2 vr[j], {j, 1, 5}];
Bound[Xi, OddTail, Delta, s] =
  Vector[DeltaStart, s].Sum[vl[j] * e[j]^3 (2/(1 - e[j]^2)^2 + 3/(1 - e[j]^2)), {j, 1, 5}] +
    Sum[(2/(1 - e[j]^2)^2 + 3/(1 - e[j]^2)) * vl[j] e[j]^3, {j, 1, 5}].Vector[DeltaEnd, s] +
    Vector[DeltaStart, s].Matrix[C, s].
      Sum[vr[j] * e[j]^2 (2/(1 - e[j]^2)^2 + 3/(1 - e[j]^2)), {j, 1, 5}] +
    Sum[(2/(1 - e[j]^2)^2 + 1/(1 - e[j]^2)) * vl[j], {j, 1, 5}].Matrix[A, s].
      (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
      Sum[2/(1 - e[j]^2) vr[j], {j, 1, 5}] +
    Sum[2/(1 - e[j]^2) * vl[j], {j, 1, 5}].Matrix[A, s].
      (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
      Sum[2/(1 - e[j]^2)^2 vr[j], {j, 1, 5}];
Do[
    Bound[XiIota, EvenTail, Delta, type, s] =
      Vector[D, type, s].Sum[vr[j] * e[j]^2 (2/(1 - e[j]^2)^2 + 2/(1 - e[j]^2)), {j, 1, 5}] +
        Sum[(2/(1 - evi[j]^2)^2 + 2/(1 - evi[j]^2)) * vi[j] * evi[j]^2, {j, 1, 5}].
          Vector[DeltaEnd, s] + Sum[2/(1 - evi[j]^2)^2 * vi[j], {j, 1, 5}].
          (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
          Sum[2/(1 - e[j]^2) vr[j], {j, 1, 5}] +
        Sum[2/(1 - evi[j]^2) * vi[j], {j, 1, 5}].
          (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
```

```
        Sum[2/(1 - e[j]^2)^2 vr[j], {j, 1, 5}];
    Bound[XiIota, OddTail, Delta, type, s] =
     Vector[D, type, s].Sum[vr[j] * e[j]^3 (2/(1 - e[j]^2)^2 + 3/(1 - e[j]^2)), {j, 1, 5}] +
      Sum[evi[j]^3 (2/(1 - evi[j]^2)^2 + 3/(1 - evi[j]^2)) * vi[j], {j, 1, 5}].
       Vector[DeltaEnd, s] + Vector[P1Iota, s].Matrix[C, s].
        Sum[vr[j] * e[j]^2 (2/(1 - e[j]^2)^2 + 3/(1 - e[j]^2)), {j, 1, 5}] +
      Sum[(2/(1 - evi[j]^2)^2 + 1/(1 - evi[j]^2)) * vi[j], {j, 1, 5}].Matrix[A, s].
       (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
        Sum[2/(1 - e[j]^2) vr[j], {j, 1, 5}] +
      Sum[2/(1 - evi[j]^2) * vi[j], {j, 1, 5}].Matrix[A, s].
       (Matrix[C, s].Matrix[A, s] + Matrix[A, s].Matrix[C, s]).
        Sum[2/(1 - e[j]^2)^2 vr[j], {j, 1, 5}];
   , {type, {0, ei}}] ×
  Clear[vl, vr, vi, e, evi]
 , {s, {i, o}}]
```

Lower bounds

The analysis of (I) requires a number of lower bounds on the coefficients. Implementing some possible lower bounds we found that while they improve our concluded bounds, using them did not allow us to lower the dimension in which we can obtain our result. So we decided to omit this lower bounds and their implications from our publication. This explains why these lower bounds are all set to zero.

```
In[4248]:= Do[
   Bound[Pi, alpha, lower, 0, s] = 0;
   Bound[Psi, lower, 0, s] = 0;
   Bound[Pi, 1, Lower, s] = 0;
   , {s, {i, o}}];
```

Bounds for differences

To the bounds stated in Lemma 5.5 of (II). For these bounds we also decided to omit a number of terms, that we could have subtracted as they did not allow us to prove the result in lower dimensions.

In[4249]:=
```
Do[
  Bound[Xi, alpha, OneMinusZero, AtZero, s] = muOverMub[s] × (Bound[Loop, 4, s]);
  Bound[Xi, alpha, ZeroMinusOne, AtZero, s] = 0;

  Bound[Xi, alpha, OneMinusZero, AtEi, s] =
   1/(2 d) muOverMub[s] × (Bound[Bubble, 3, s] + 2 Bound[Loop, 4, s]);
  Bound[Xi, alpha, ZeroMinusOne, AtEi, s] = 0;
  Bound[Psi, alphaI, OneMinusZero, AroundEi, s] =
   2 gjz[s]^2 (Bound[tG, {0, 1}, 4, s] + (2 d - 2) Bound[tG, {2}, 2, s]) +
    (2 d - 2) gjz[s]^2 (2 d gjz[s])^2 VarGamma2[s] × K[2, 2, {2}] +
    gjz[s]^2 (2 d gjz[s])^4 VarGamma2[s] × K[2, 4, {0, 1}];

  Bound[Psi, alphaI, ZeroMinusOne, AroundEi, s] =
   2 gjz[s]^2 (Bound[tG, {0, 1}, 4, s] + (2 d - 2) Bound[tG, {2}, 2, s]);

  Bound[Psi, alphaII, OneMinusZero, AroundZero, s] =
   (2 d - 1) Bound[Xi, alpha, OneMinusZero, AtEi, s];

  Bound[Psi, alphaII, ZeroMinusOne, AroundZero, s] =
   (2 d - 1) Bound[Xi, alpha, ZeroMinusOne, AtEi, s];
  , {s, {i, o}}];
```

Summing the bounds

We compute the sum over all odd/even N.

In[4250]:=
```
Do[
 Bound[Xi, Even, s] = Sum[Bound[Xi, t, s], {t, {0, 2, EvenTail}}];
 Bound[Xi, Odd, s] = Sum[Bound[Xi, t, s], {t, {1, 3, OddTail}}];
 Bound[Xi, Absolut, s] = Sum[Bound[Xi, t, s], {t, {Odd , Even}}];

 Bound[Xi, Even, Delta, s] = Sum[Bound[Xi, t, Delta, s], {t, {0, 2, EvenTail}}];
 Bound[Xi, Odd, Delta, s] = Sum[Bound[Xi, t, Delta, s], {t, {1, 3, OddTail}}];
 Bound[Xi, Absolut, Delta, s] = Sum[Bound[Xi, t, Delta, s], {t, {Odd , Even}}];


 Bound[XiIota, Even, s] = Sum[Bound[XiIota, t, s], {t, {0, 2, EvenTail}}];
 Bound[XiIota, Odd, s] = Sum[Bound[XiIota, t, s], {t, {1, 3, OddTail}}];
 Bound[XiIota, Absolut, s] = Sum[Bound[XiIota, t, s], {t, {Odd , Even}}];

 Do[
  Bound[XiIota, Even, Delta, type, s] =
   Sum[Bound[XiIota, t, Delta, type, s], {t, {0, EvenTail}}];
  Bound[XiIota, Odd, Delta, type, s] =
   Sum[Bound[XiIota, t, Delta, type, s], {t, {1, OddTail}}];
  Bound[XiIota, Absolut, Delta, type, s] =
   Sum[Bound[XiIota, t, Delta, type, s], {t, {Odd, Even}}];
  , {type, {ei, 0}}];

 Clear[type, t];
 , {s, {i, o}}]
```

Bound on the simplified rewrite

In the preceding section we have computed all bounds required by Assumption 4.3 of (I). We use the methods provided in the General.nb-Notebook to compute the bounds on the simplified rewrite, as derived in Appendix D of (I).

In[4251]:=
```
Do[
 mu[s] = gjz[s];
 mub[s] = gz[s];
 mumin[s] = 1/(2 d - 1);

 beta[CPhi, Lower, s] =
  betaCPhiLow[d, mu[s], Bound[Xi, alpha, OneMinusZero, AtZero, s],
   Bound[XiIota, alphaI, 0, Atei, s]];
 beta[CPhi, Upper, s] =
  betaCPhiUp[d, mu[s], Bound[Xi, alpha, ZeroMinusOne, AtZero, s],
   Bound[XiIota, alphaII, 0, AtZero, s]];

 beta[af, Lower, s] = betaAfLow[d, mumin[s], mu[s],
   Bound[Psi, alphaI, OneMinusZero, AroundEi, s],
   Bound[Psi, alphaII, ZeroMinusOne, AroundZero, s], Bound[Pi, alpha, 0, s]];
```

```
beta[af, Upper, s] =
 betaAfUp[d, mu[s], Bound[Psi, alphaI, ZeroMinusOne, AroundEi, s],
  Bound[Psi, alphaII, OneMinusZero, AroundZero, s],
  Bound[Pi, alpha, lower, 0, s]];

beta[ap, s] = betaap[d, mu[s], Bound[Xi, alpha, OneMinusZero, AtEi, s],
  Bound[Xi, alpha, ZeroMinusOne, AtEi, s],
  Bound[XiIota, alphaI, 0, SumAroundei, s],
  Bound[XiIota, alphaII, 0, SumAroundZero, s]];

beta[PiHat, s] = betaPiHat[d, mub[s], Bound[XiIota, Even, s],
  Bound[Pi, 1, Lower, s]];
beta[PsiHat, s] = betaPsiHatLower[d, mubOverMu[s], Bound[Xi, Odd, s],
  Bound[Psi, lower, 0, s]];


beta[Rf, s] = betaRF[d, mu[s], mub[s], mubOverMu[s], mub[s],
  Bound[Xi, Absolut, s], Bound[Xi, EvenTail, s] + Bound[Xi, OddTail, s],
  Bound[XiIota, Absolut, s], Sum[Bound[XiIota, t, s], {t, {Odd , EvenTail}}],
  Bound[Psi, RI, 0, s] + Bound[Psi, RI, 1, s],
  Bound[Psi, RII, 0, s] + Bound[Psi, RII, 1, s], Bound[Pi, R, 0, s]];


beta[Rp, s] = betaRp[d, mu[s], mubOverMu[s], mub[s], Bound[Xi, Absolut, s],
  Bound[Xi, R, 0, s] + Bound[Xi, R, 1, s],
  Bound[Xi, EvenTail, s] + Bound[Xi, OddTail, s], Bound[XiIota, Absolut, s],
  Sum[Bound[XiIota, t, s], {t, {Odd , EvenTail}}], Bound[XiIota, RI, 0, s],
  Bound[XiIota, RII, 0, s]];

beta[Rp, Delta, s] = betaRpDelta[d, mu[s], mubOverMu[s], mub[s],
  Bound[Xi, Absolut, s], Bound[Xi, Absolut, Delta, s],
  Bound[Xi, R, 0, s] + Bound[Xi, R, 1, s],
  Sum[Bound[Xi, t, Delta, s], {t, {OddTail , EvenTail}}],
  Bound[XiIota, Absolut, s], Bound[XiIota, Absolut, Delta, ei, s],
  Bound[XiIota, Absolut, Delta, 0, s],
  Sum[Bound[XiIota, t, Delta, ei, s], {t, {Odd, EvenTail}}],
  Sum[Bound[XiIota, t, Delta, 0, s], {t, {Odd, EvenTail}}],
  Bound[XiIota, RI, 0, Delta, ei, s], Bound[XiIota, RII, 0, Delta, 0, s]];


beta[Rf, abs, Delta, s] = betaRfDelta[d, mu[s], mubOverMu[s], mub[s],
  Bound[Xi, Absolut, s], Bound[Xi, Absolut, Delta, s],
  Bound[Xi, EvenTail, s] + Bound[Xi, OddTail, s],
  Sum[Bound[Xi, t, Delta, s], {t, {OddTail , EvenTail}}],
  Bound[Psi, RI, 0, Delta, s] + Bound[Psi, RI, 1, Delta, s],
  Bound[Psi, RII, 0, Delta, s] + Bound[Psi, RII, 1, Delta, s],
```

```
    Bound[XiIota, Absolut, s], Bound[XiIota, Absolut, Delta, ei, s],
    Bound[XiIota, Absolut, Delta, 0, s],
    Sum[Bound[XiIota, t, s], {t, {Odd, EvenTail}}],
    Sum[Bound[XiIota, t, Delta, ei, s], {t, {Odd, EvenTail}}],
    Bound[Pi, R, 0, Delta, eiek, s]];

  beta[Rf, Lower, Delta, s] =
   betaRfDeltaLower[d, mu[s], mubOverMu[s], mub[s], Bound[Xi, Absolut, s],
    Bound[Xi, Odd, s], Bound[Xi, Even, s], Bound[Xi, Absolut, Delta, s],
    Bound[Xi, Odd, Delta, s], Bound[Xi, Even, Delta, s], Bound[Xi, OddTail, s],
    Bound[Xi, OddTail, Delta, s], Bound[Xi, EvenTail, Delta, s],
    Bound[Psi, RI, 1, Delta, s], Bound[Psi, RII, 0, Delta, s],
    Bound[XiIota, Absolut, s], Bound[XiIota, Odd, s], Bound[XiIota, Even, s],
    Bound[XiIota, Absolut, Delta, ei, s], Bound[XiIota, Odd, Delta, ei, s],
    Bound[XiIota, Even, Delta, ei, s], Bound[XiIota, Absolut, Delta, 0, s],
    Bound[XiIota, Odd, Delta, 0, s], Bound[XiIota, Odd, Delta, 0, s],
    Bound[XiIota, EvenTail, s], Bound[XiIota, EvenTail, Delta, ei, s],
    Bound[Pi, R, 0, Delta, eiek, s]];
  , {s, {i, o}}]
```

In[4252]:= `beta[Rf, abs, Delta, o]`

Out[4252]= `0.10325`

Improvement of Bounds

In this section we implement the computations of Section 3 of (I) to verify whether we can conclude from $f_i(z) \leq \Gamma_i$ that $f_i(z) < \Gamma_i - \epsilon$. The sufficient condition for this to succeed is stated in Definition 2.9 of (I). We check the conditions one line at a time.

Technical conditions

All these conditions are necessary conditions. However, numerically they are next to trivial, in the sense that other conditions (most likely f_2 or f_3) will fail first

In[4253]:=
```
Do[
 TechCondition[I, s] = (beta[CPhi, Lower, s] - beta[ap, s] - beta[Rp, s]) > 0;
 (* Part of Assumption 2.7. of (I), stating that the nominator of Ĝz(k),
 being Φ̂(k), is positive *)
 TechCondition[II, s] = (beta[af, Lower, s] - beta[Rf, Lower, Delta, s]) > 0;
 (*Part of Assumption 2.7. of (I),
 neccesary to ensure that f2 is well defined *)
 TechCondition[III, s] = ((2 d - 1) mub[s] / (1 - mu[s]) Bound[XiIota, Absolut, s]) < 1;
 (* Condition (4.34) of (I),
 which is necessary to make the geometric series converge *)
 TechCondition[IV, s] =
  N[z[i]^4 (1 + (2 d - 2) z[i])^4] - (2 × (2 d - 2) gjz[s]^4 + Bound[tG, {2}, 6, s])^2 > 0;
 (* For lattice animals we verify that Ξ^(0),ι(eι+eκ)>Π^(0),ι,κ(eι+eκ),
 see (3.70) of (II).  Starting from (3.70) we derive a simpler numerical
  condition, that easier to verify. All details are explained below
  TODO Monoton?*)

 TechCondition[s] = TechCondition[I, s] && TechCondition[II, s] &&
   TechCondition[III, s] && TechCondition[IV, s]
 , {s, {i, o}}]
```

Improvement of f_1

Following are the bounds derived in Section 3.1 of (I):

In[4254]:=
```
Do[

 boundF1[part1, s] = mubOverMu[s] (1 + beta[PiHat, s]) / (1 - (2 d)/(2 d-1) beta[PsiHat, s]);

 boundF1[part2, s] = cmu (1 + beta[PiHat, s]) / (1 - (2 d)/(2 d-1) beta[PsiHat, s]);

 boundF1[s] = Max[boundF1[part1, s], boundF1[part2, s]];

 , {s, {i, o}}]
```

Improvement of f_2

Next we implement the bound derived in Section 3.2 of (I)

In[4255]:=
```
Do[
 boundF2[s] = (2 d - 1)/(2 d - 2) (beta[CPhi, Upper, s] + beta[ap, s] + beta[Rp, s]) / (beta[af, Lower, s] + beta[Rf, Lower, Delta, s]);
 , {s, {i, o}}]
```

Improvement of f_3

Here we start with some preparation and and re-implement the bounds on H_1 as given in (3.71) of (I) to using only monotone bounds.

In[4256]:=

```
BoundH[1, d_, n_, l_, v_, Gamma2dash_, cp_, afmin_, afmax_, ap_, bRf_,
   bRp_, bRfDelta_, bRpDelta_, Kunderline_] :=
  If[n == 0, (* (3.61)  *) cp Jnl[v, 0, l] + ap Jnl[v, 0, l + 1] + ap/afmin Jnl[v, -1, l]
   (*IM[-1,l,v] is defined accordingly *), If[n == 1, (* (3.64) *)
    cp^2/afmin Jnl[v, 1, l] + (cp ap)/afmin Jnl[v, 0, l] + 2 (ap cp)/afmin Jnl[v, 1, l + 1] +
     ap^2/afmin Jnl[v, 0, l + 1] + ap^2/afmin Jnl[v, 1, l + 2] +
     (bRp + bRfDelta Gamma2dash)/afmin^2 (cp T[3, l, v] + ap T[3, l + 1, v] + ap T[2, l, v]),
    If[n == 2, (* (3.68)+(3.70) *)
     cp^3/afmin^2 Jnl[v, 2, l] + (cp^2 ap)/afmin^2 Jnl[v, 1, l] + 3 (ap cp^2)/afmin^2 Jnl[v, 2, l + 1] +
      (2 cp^2 ap)/afmin^2 Jnl[v, 1, l + 1] + 3 (ap^2 cp)/afmin^2 Jnl[v, 2, l + 2] + ap^3/afmin^2 Jnl[v, 1, l + 2] +
      ap^3/afmin^2 Jnl[v, 2, l + 3] + (bRp + bRfDelta Gamma2dash)/afmin^2 (cp/afmin + Gamma2dash)
       (cp T[4, l, v] + ap T[4, l + 1, v] + ap T[3, l, v]) +
      ap (bRp + bRfDelta Gamma2dash)/afmin^3
       ( cp T[4, l + 1, v] + ap T[4, l + 2, v] + ap T[3, l + 1, v]), -1]]];
BoundFThree[d_, n_, l_, vecs_, Gamma2dash_, cp_, afmin_, afmax_, ap_,
   bRf_, bRp_, bRfDelta_, bRpDelta_, Kunderline_] := Module[{v2, i},
   v2 = Table[BoundH[d, n, l, vecs[[i]], Gamma2dash, cp, afmin, afmax, ap,
      bRf, bRp, bRfDelta, bRpDelta, Kunderline], {i, 1, Length[vecs]}];
   Max[v2]
  ];
```

In[4258]:=

```
(* The values to compare to *)
const[1] = GammaThreeClosed[1, 4];
const[2] = GammaThreeClosed[2, 4];
const[3] = GammaThree[1, 0];
const[4] = GammaThree[1, 1];
const[5] = GammaThree[1, 2];
const[6] = GammaThree[1, 3];
const[7] = GammaThree[2, 0];
const[8] = GammaThree[2, 1];
const[9] = GammaThree[2, 2];
const[10] = GammaThree[2, 3];

(* Initial bounds as in Section 3.3.3, obtained by SRW computations,
Methods provided in General.nb *)
boundF3[1, i] = BoundFThreeInital[d, 1, 6, 1, {{0}} ];
boundF3[2, i] = BoundFThreeInital[d, 2, 6, 1, {{0}} ];

boundF3[3, i] = BoundFThreeInital[d, 1, 0, 1, {{1}, {2}, {0, 1}} ];
boundF3[4, i] = BoundFThreeInital[d, 1, 1, 1, {{1}, {2}, {0, 1}} ];
boundF3[5, i] = BoundFThreeInital[d, 1, 2, 1, {{1}, {2}, {0, 1}} ];
boundF3[6, i] = BoundFThreeInital[d, 1, 3, 1, {{1}, {2}, {0, 1}} ];

boundF3[7, i] = BoundFThreeInital[d, 2, 0, 1, {{1}, {2}, {0, 1}} ];
boundF3[8, i] = BoundFThreeInital[d, 2, 1, 1, {{1}, {2}, {0, 1}} ];
boundF3[9, i] = BoundFThreeInital[d, 2, 2, 1, {{1}, {2}, {0, 1}} ];
boundF3[10, i] = BoundFThreeInital[d, 2, 3, 1, {{1}, {2}, {0, 1}} ];
```

In[4278]:=

In[4279]:=
```
(* The bounds summarized in (3.80)of (I) *)
BoundFThreeBound[n_, l_, vs_] :=
  BoundFThree[d, n, l, vs, Gamma2, beta[CPhi, Upper, o], beta[af, Lower, o],
   beta[af, Upper, o], beta[ap, o], beta[Rp, o], beta[Rf, o],
   beta[Rf, abs, Delta, o], beta[Rp, Delta, o],
   1 / (beta[af, Lower, o] + beta[Rf, Lower, Delta, o])];

boundF3[1, o] := BoundFThreeBound[1, 6, {{0}}];
boundF3[2, o] = BoundFThreeBound[2, 6, {{0}}];

BoundFThreeOpen[n_, l_] := BoundFThreeBound[n, l, {{1}, {2}, {0, 1}}];

onstepContribution = BoundFThreeBound[1, 2, {{1}}] +
   1/(2 d) (2 × (2 d - 2) Bound[tG, {2}, 2, o] + 4 Bound[tG, {0, 1}, 2, o]);
boundF3[3, o] = Max[2 d gjz[o] onstepContribution + Bound[tG, {1}, 1, o],
   2 d gjz[o] × BoundFThreeBound[1, 1, {{2}}] + Bound[tG, {2}, 2, o],
   2 d gjz[o] × BoundFThreeBound[1, 1, {{0, 1}}] + Bound[tG, {0, 1}, 2, o],
   BoundFThreeBound[1, 0, {{2}, {0, 1}}]];
boundF3[4, o] = Max[onstepContribution, BoundFThreeBound[1, 1, {{2}, {0, 1}}]];
boundF3[5, o] = BoundFThreeOpen[1, 2];
boundF3[6, o] = BoundFThreeOpen[1, 3];

boundF3[7, o] =
  Max[2 d gjz[o] × (BoundFThreeBound[1, 1, {{1}}] + BoundFThreeBound[2, 1, {{1}}]) +
     Bound[tG, {1}, 2, o],
    2 d gjz[o] × (BoundFThreeBound[1, 1, {{2}}] + BoundFThreeBound[2, 1, {{2}}]) +
     Bound[tG, {2}, 2, o],
    2 d gjz[o] × (BoundFThreeBound[1, 1, {{0, 1}}] +
        BoundFThreeBound[2, 1, {{0, 1}}]) + + Bound[tG, {0, 1}, 2, o],
    BoundFThreeBound[2, 0, {{2}, {0, 1}}]];
boundF3[8, o] = BoundFThreeOpen[2, 1];
boundF3[9, o] = BoundFThreeOpen[2, 2];
boundF3[10, o] = BoundFThreeOpen[2, 3];


BoundsF3Table = Table[boundF3[j, s] / const[j], {s, {i, o}}, {j, 1, 10}];
boundF3[i] = Ceiling[Max[BoundsF3Table[[1]]], 10^-9];
boundF3[o] = Max[BoundsF3Table[[2]]];
```

Results

Preparation of output

```
In[4294]:= Do[
   SuccesF[1, s] = boundF1[s] < Gamma1;
   SuccesF[2, s] = boundF2[s] < Gamma2;
   SuccesF[3, s] = boundF3[s] < Gamma3;
   Succes[s] = SuccesF[1, s] && SuccesF[2, s] && SuccesF[3, s] && TechCondition[s];
   , {s, {i, o}}];
 Succes[overall] = Succes[i] && Succes[o];

In[4296]:= overAllStatement = "The statement that the bootstrap was successful is "
    If[Succes[overall], Style[TextString[Succes[overall]], Bold, Green],
     Style[TextString[Succes[overall]], Bold, Red]];

In[4297]:= bubbles = {Graphics[{Green, Disk[{0, 0}, 0.8]}, ImageSize → 30],
    Graphics[{Red, Disk[{0, 0}, 0.8]}, ImageSize → 40]};

 Do[
  CoefficientboundsTable[s] =
   {{Quantity, "Ξ^Zero", "Ξ^One", "Ξ^Two", "Ξ^Three", "Ξ^Even,>3", "Ξ^Odd,>3"},
    {Text[Bound for Ξ̂], Bound[Xi, 0, s], Bound[Xi, 1, s] , Bound[Xi, 2, s],
     Bound[Xi, 3, s] , Bound[Xi, EvenTail, s], Bound[Xi, OddTail, s]},
    {Text[Bound for Ξ̂^ι], Bound[XiIota, 0, s], Bound[XiIota, 1, s] ,
     Bound[XiIota, 2, s], Bound[XiIota, 3, s] , Bound[XiIota, EvenTail, s],
     Bound[XiIota, OddTail, s]},
    {Text[Ξ̂ "|x|_2^2" ], Bound[Xi, 0, Delta, s], Bound[Xi, 1, Delta, s],
     Bound[Xi, 2, Delta, s], Bound[Xi, 3, Delta, s], Bound[Xi, EvenTail, Delta, s],
     Bound[Xi, OddTail, Delta, s]},
    {Text[Ξ^ι "|x-e_ι|_2^2"] , Bound[XiIota, 0, Delta, ei, s],
     Bound[XiIota, 1, Delta, ei, s]  , Bound[XiIota, 2, Delta, ei, s],
     Bound[XiIota, 3, Delta, ei, s], Bound[XiIota, EvenTail, Delta, ei, s],
     Bound[XiIota, OddTail, Delta, ei, s]},
    {Text[Ξ^ι "|x|_2^2"] , Bound[XiIota, 0, Delta, 0, s], Bound[XiIota, 1, Delta, 0, s]  ,
     Bound[XiIota, 2, Delta, 0, s], Bound[XiIota, 3, Delta, 0, s],
     Bound[XiIota, EvenTail, Delta, 0, s], Bound[XiIota, OddTail, Delta, 0, s]}};
  TableCoefficients[s] =
   Labeled[Grid[CoefficientboundsTable[s], Alignment → {Center},
      Frame → True, Dividers → {{2 → True, -1 → True}, {2 → True}},
      ItemStyle → {1 → Bold, 1 → Bold},
      Background → {{None}, {GrayLevel[0.9]}, {None}}],
     Style["Bounds on the coefficients in dimension in " <> TextString[d], Bold],
     Top] // Text;


  CoefficientSplitsBoundsI[s] =
   {{Quantity, "Ξ_R^Zero", "Ψ_RI^Zero", "Ψ_RII^Zero", "Ξ_R^One", "Ψ_RI^One", "Ψ_RII^One"},
```

```
  {Text[Abs Bound], Bound[Xi, R, 0, s], Bound[Psi, RI, 0, s],
   Bound[Psi, RII, 0, s], Bound[Xi, R, 1, s] , Bound[Psi, RI, 1, s],
   Bound[Psi, RII, 1, s]}, {Text[Ξ̂ "|x|_2^2" ], Bound[Xi, R, 0, Delta, s],
   Bound[Psi, RI, 0, Delta, s], Bound[Psi, RII, 0, Delta, s],
   Bound[Xi, R, 1, Delta, s] , Bound[Psi, RI, 1, Delta, s],
   Bound[Psi, RII, 1, Delta, s]}};
TableCoefficientSplitI[s] =
 Labeled[Grid[CoefficientSplitsBoundsI[s], Alignment → {Center},
    Frame → True, Dividers → {{2 → True, -1 → True}, {2 → True}},
    ItemStyle → {1 → Bold, 1 → Bold},
    Background → {{None}, {GrayLevel[0.9]}, {None}}],
   Style["Bounds on the remainder term of the split of Ξ and Ψ in d=" <>
     TextString[d], Bold], Top] // Text;

CoefficientSplitsBoundsII[s] =
 {{Quantity, " Ξ̂^zero_alpha(0)-Ξ̂^One_alpha(0)", " Ξ̂^zero_alpha(e_1)-Ξ̂^One_alpha(e_1)", "∑Ψ^Zero_alphaI-Ψ^One_alphaI",
   "∑Ψ^Zero_alphaII-Ψ^One_alphaII"},
  {Text[Lower Bound], Bound[Xi, alpha, OneMinusZero, AtZero, s],
   Bound[Xi, alpha, OneMinusZero, AtEi, s],
   Bound[Psi, alphaI, OneMinusZero, AroundEi, s],
   Bound[Psi, alphaII, OneMinusZero, AroundZero, s]},
  {Text[Upper Bound ], Bound[Xi, alpha, ZeroMinusOne, AtZero, s],
   Bound[Xi, alpha, ZeroMinusOne, AtEi, s],
   Bound[Psi, alphaI, ZeroMinusOne, AroundEi, s],
   Bound[Psi, alphaII, ZeroMinusOne, AroundZero, s]}};
TableCoefficientSplitII[s] =
 Labeled[Grid[CoefficientSplitsBoundsII[s], Alignment → {Center},
    Frame → True,
    Dividers → {{2 → True, 3 → True, 4 → True, 5 → True, -1 → True}, {2 → True}},
    ItemStyle → {1 → Bold, 1 → Bold},
    Background → {{None}, {GrayLevel[0.9]}, {None}}],
   Style["Bounds on the difference of explicit terms in dimension " <>
     TextString[d], Bold], Top] // Text;

CoefficientSplitsBoundsIII[s] =
 {{Quantity, " Ξ̂^ι", "|x-e_ι|_2^2 Ξ̂^ι", " Ξ̂^ι_alphaI(e_ι)", "∑ Ξ̂^ι_alphaI", " Ξ̂^ι_RI",
   "|x-e_ι|_2^2 Ξ̂^ι_RI", "|x|_2^2 Ξ̂^ι", " Ξ̂^ι_alphaII(0)", "∑ Ξ̂^ι_alphaII", "Ξ̂^ι_RII", "|x|_2^2 Ξ̂^ι_RII"},
  {, Bound[XiIota, 0, s], Bound[XiIota, 0, Delta, ei, s],
   Bound[XiIota, alphaI, 0, Atei, s], Bound[XiIota, alphaI, 0, SumAroundei, s],
   Bound[XiIota, RI, 0, s], Bound[XiIota, RI, 0, Delta, ei, s],
   Bound[XiIota, 0, Delta, 0, s], Bound[XiIota, alphaII, 0, AtZero, s],
   Bound[XiIota, alphaII, 0, SumAroundZero, s], Bound[XiIota, RII, 0, s],
   Bound[XiIota, RII, 0, Delta, 0, s]}};
TableCoefficientSplitIII[s] =
 Labeled[Grid[CoefficientSplitsBoundsIII[s], Alignment → {Center},
```

```
      Frame → True, Dividers → {{2 → True, 5 → True, 8 → True, -1 → True}, {2 → True}},
      ItemStyle → {1 → Bold, 1 → Bold},
      Background → {{None}, {GrayLevel[0.9]}, {None}}],
     Style["Bounds on split of the coefficient Ξ(0),ι in dimension " <>
       TextString[d], Bold], Top] // Text;
 , {s, {i, o}}]

SimpleNotationBoundsPart1 =
  {{Quantity, "aF-Lower", "aF-Upper", "|aΦ|", "cΦ-Lower", "cΦ-Upper", Π, Ψ},
   {"Bound i" , beta[af, Lower, i], beta[af, Upper, i], beta[ap, i],
    beta[CPhi, Lower, i], beta[CPhi, Upper, i], beta[PiHat, i], beta[PsiHat, i]},
   {"Bound o" , beta[af, Lower, o], beta[af, Upper, o], beta[ap, o],
    beta[CPhi, Lower, o], beta[CPhi, Upper, o], beta[PiHat, o], beta[PsiHat, o]}};

TableSimpleNotation1 =
  Labeled[Grid[SimpleNotationBoundsPart1, Alignment → {Center},
      Frame → True, Dividers → {{2 → True, -1 → True}, {2 → True}},
      ItemStyle → {1 → Bold, 1 → Bold},
      Background → {{None}, {GrayLevel[0.9]}, {None}}],
     Style["Bounds on simplified rewrite in dimension " <> TextString[d], Bold],
     Top] // Text;

SimpleNotationBoundsPart2 =
  {{Quantity, "|RF|", "|RΦ|", "x2^2RF-lower", "|x2^2RF|", "|x2^2RΦ|"},
   {"Bound i", beta[Rf, i], beta[Rp, i], beta[Rf, Lower, Delta, i],
    beta[Rf, abs, Delta, i], beta[Rp, Delta, i]},
   {"Bound o", beta[Rf, o], beta[Rp, o], beta[Rf, Lower, Delta, o],
    beta[Rf, abs, Delta, o], beta[Rp, Delta, o]}};
TableSimpleNotation2 =
  Labeled[Grid[SimpleNotationBoundsPart2, Alignment → {Center},
      Frame → True, Dividers → {{2 → True, 4 → True, -1 → True}, {2 → True}},
      ItemStyle → {1 → Bold, 1 → Bold},
      Background → {{None}, {GrayLevel[0.9]}, {None}}],
     Style["Bounds on remainder of rewrite in dimension " <> TextString[d], Bold],
     Top] // Text;

ContentCheckf1f2 =
  {{Bounds, "f1(zI)", "f2(zI)", "f3(zI)", "f1(z)", "f2(z)", "f3(z)"},
   {"Γi", NumberForm[Gamma1, 10], NumberForm[Gamma2, 11], NumberForm[Gamma3, 11],
    NumberForm[Gamma1, 10], NumberForm[Gamma2, 10], NumberForm[Gamma3, 10] },
   {Bounds, NumberForm[boundF1[i], 10], NumberForm[boundF2[i], 10],
    NumberForm[N[boundF3[i]], 10], NumberForm[boundF1[o], 10],
    NumberForm[boundF2[o], 10], NumberForm[boundF3[o], 10]}, { "check",
    If[SuccesF[1, i], bubbles[[1]], bubbles[[2]]],
    If[SuccesF[2, i], bubbles[[1]], bubbles[[2]]],
    If[SuccesF[3, i], bubbles[[1]], bubbles[[2]]],
```

```
    If[SuccesF[1, o], bubbles[[1]], bubbles[[2]]],
    If[SuccesF[2, o], bubbles[[1]], bubbles[[2]]],
    If[SuccesF[3, o], bubbles[[1]], bubbles[[2]]]}};

TableCheckf1f2 =
  Labeled[Grid[ContentCheckf1f2, Alignment → {Center}, Frame → True,
     Dividers → {{2 → True, -1 → True}, {2 → True}},
     ItemStyle → {1 → Bold, 1 → Bold},
     Background → {{None}, {GrayLevel[0.9]}, {None}}],
    Style["Bounds on the LA-bootstrap functions in dimension " <> TextString[d],
     Bold], Top] // Text;

Do[
 tableCheckf3Bubble[s] =
  {{Bounds, "F3-1,6,{0}", "F3-1,0,x≠0", "F3-1,1,x≠0", "F3-1,2,x≠0", "F3-1,3,x≠0"},
   {"Assumed bound", N[const[1] * Gamma3], NumberForm[const[3] * Gamma3, 10],
    NumberForm[const[4] * Gamma3, 10], NumberForm[const[5] * Gamma3, 10],
    NumberForm[const[6] * Gamma3, 10] },
   {"Concluded bound", boundF3[1, s], boundF3[3, s], boundF3[4, s],
    boundF3[5, s], boundF3[6, s]}, { "check",
    If[boundF3[1, s] < const[1] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[3, s] < const[3] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[4, s] < const[4] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[5, s] < const[5] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[6, s] < const[6] * Gamma3, bubbles[[1]], bubbles[[2]]]}};

 tableCheckf3Triangle[s] =
  {{Bounds, "F3-2,6,{0}", "F3-2,0,x≠0", "F3-2,1,x≠0", "F3-2,2,x≠0", "F3-2,3,x≠0"},
   {"Assumed bound", N[const[2] * Gamma3, 10] , NumberForm[const[7] * Gamma3, 10],
    NumberForm[const[8] * Gamma3, 10], NumberForm[const[9] * Gamma3, 10],
    NumberForm[const[10] * Gamma3, 10] },
   {"Concluded bound", boundF3[2, s], boundF3[7, s], boundF3[8, s],
    boundF3[9, s], boundF3[10, s]}, { "check",
    If[boundF3[2, s] < const[2] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[7, s] < const[7] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[8, s] < const[8] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[9, s] < const[9] * Gamma3, bubbles[[1]], bubbles[[2]]],
    If[boundF3[10, s] < const[10] * Gamma3, bubbles[[1]], bubbles[[2]]]}};

 , {s, {i, o}}]

TableCheckf3Bubble =
  Labeled[Grid[tableCheckf3Bubble[o], Alignment → {Center}, Frame → True,
     Dividers → {{2 → True, -1 → True}, {2 → True}},
     ItemStyle → {1 → Bold, 1 → Bold},
     Background → {{None}, {GrayLevel[0.9]}, {None}}],
    Style["Bounds on the LA-weighted bubble in dimension " <> TextString[d],
     Bold], Top] // Text;
```

```
TableCheckf3Triangle =
  Labeled[Grid[tableCheckf3Triangle[o], Alignment → {Center}, Frame → True,
     Dividers → {{2 → True, -1 → True}, {2 → True}},
     ItemStyle → {1 → Bold, 1 → Bold},
     Background → {{None}, {GrayLevel[0.9]}, {None}}],
    Style["Bounds on the LA-weighted triangle in dimension " <> TextString[d],
     Bold], Top] // Text;
```

Simple output

```
In[4308]:= overAllStatement

Out[4308]= The statement that the bootstrap was successful is True
```

If this succeeds, then the infrared bound Theorem 2.10 of (I) or Theorem 1.1 of (II) holds with

```
In[4309]:= "Ĝz(k)[1-D̂(k)] ≤ " <> TextString[Ceiling[(2 d - 2)/(2 d - 1) Gamma2, 0.001]]
           "Ĝ̂z(k)[1-D̂(k)] ≤ " <> TextString[Ceiling[g[o] (2 d - 2)/(2 d - 1) Gamma2, 0.001]] <>
            " = A2(d)"
           "A1(d) = " <>
            TextString[
             Ceiling[g[o] × (beta[CPhi, Upper, o] + beta[ap, o] + beta[Rp, o])
               Max[1/(beta[af, Lower, o] + beta[Rf, Lower, Delta, o]),
                1/(beta[CPhi, Lower, o] - beta[ap, o] - beta[Rp, o])], 0.001]]

Out[4309]= Ĝz(k)[1-D̂(k)] ≤ 1.106

Out[4310]= Ĝ̂z(k)[1-D̂(k)] ≤ 3.187 = A2(d)

Out[4311]= A1(d) = 3.185
```

Further, we have proven that g_{z_c} and $g_{z_c} z_c$, respectively are smaller than:

```
In[4312]:= Print["g_{z_c} ≤ ", Ceiling[Exp[1] Gamma1, 0.0001]]
           Print["(2d-1) z_c g_{z_c} ≤ ", Ceiling[Gamma1, 0.0001]]

   g_{z_c} ≤ 2.8814
   (2d-1) z_c g_{z_c} ≤ 1.06

In[4314]:= 5749 / 2 000 000

Out[4314]= 5749/2 000 000
```

In[4315]:= $N\left[\frac{5749}{2\,000\,000}\right]$

Out[4315]= 0.0028745

The bounds on the bootstrap functions

In[4316]:=
```
TableCheckf1f2
TableCheckf3Bubble
TableCheckf3Triangle
```

Out[4316]=

Bounds on the LA–bootstrap functions in dimension 30

Bounds	$f_1(z_I)$	$f_2(z_I)$	$f_3(z_I)$	$f_1(z)$	$f_2(z)$	$f_3(z)$
Γ_i	1.06	1.125	1	1.06	1.125	1
Bounds	1.055026518	1.07166728	0.477109839	1.057166397	1.124107039	0.9906769662
check						

Out[4317]=

Bounds on the LA–weighted bubble in dimension 30

Bounds	F3–1,6,{0}	F3–1,0,x≠0	F3–1,1,x≠0	F3–1,2,x≠0	F3–1,3,x≠0
Assumed bound	0.005	0.05	0.007	0.003	0.001
Concluded bound	0.0041709	0.0437427	0.00658719	0.00266199	0.000932884
check					

Out[4318]=

Bounds on the LA–weighted triangle in dimension 30

Bounds	F3–2,6,{0}	F3–2,0,x≠0	F3–2,1,x≠0	F3–2,2,x≠0	F3–2,3,x≠0
Assumed bound	0.012	0.7	0.016	0.005	0.002
Concluded bound	0.0104912	0.0603389	0.0146566	0.00440207	0.00198135
check					

Bounds of the coefficients bounds

In[4319]:=
```
TableCoefficients[o]
TableSimpleNotation1
TableSimpleNotation2
TableCoefficientSplitI[o]
TableCoefficientSplitII[o]
TableCoefficientSplitIII[o]
```

Bounds on the coefficients in dimension in 30

Out[4319]=

Quantity	Ξ^{Zero}	Ξ^{One}	Ξ^{Two}	Ξ^{Three}	$\Xi^{Even,>3}$	$\Xi^{Odd,>3}$
Bound for $\hat{\Xi}$	0.00038824	0.00401534	0.0000580677	7.74458×10^{-7}	1.15924×10^{-8}	1.6358×10^{-10}
Bound for $\hat{\Xi}'$	0.0177534	0.00018869	2.72782×10^{-6}	3.63867×10^{-8}	5.43952×10^{-10}	7.67662×10^{-12}
$\lvert x\rvert_2^2\,\hat{\Xi}$	0.00409038	0.0453281	0.029782	0.000820774	0.0000445553	0.0000185923
$\lvert x-e_I\rvert_2^2\,\Xi'$	0.000100318	0.00376966	0.00149271	0.0000401392	2.10867×10^{-6}	4.17787×10^{-8}
$\lvert x\rvert_2^2\,\Xi'$	0.000100184	0.00261601	0.00144116	0.000039229	2.08689×10^{-6}	4.13946×10^{-8}

Bounds on simplified rewrite in dimension 30

Out[4320]=

Quantity	a_F–Lower	a_F–Upper	$\lvert a_\Phi\rvert$	c_Φ–Lower	c_Φ–Upper	Π	Ψ
Bound i	1.01700939291119722142979252245376 28	1.01728615978857042739226927208189 155	0.018961	0.999336	1.00029761711667738542492317514397 4046	0.0177608	0.00342959315424924786170957255980 7
Bound o	1.01679	1.0439	0.0201073	0.999237	1.00032	0.0191405	0.00408734

Bounds on remainder of rewrite in dimension 30

Out[4321]=

Quantity	$\lvert R_F\rvert$	$\lvert R_\Phi\rvert$	$x_2^2R_F$–lower	$\lvert x_2^2R_F\rvert$	$\lvert x_2^2R_\Phi\rvert$
Bound i	0.0586935	0.00234883	−0.0472855	0.0508183	0.00290749
Bound o	0.0650546	0.0028833	−0.0907562	0.10325	0.00811684

Bounds on the remainder term of the split of Ξ and Ψ in d=30

Out[4322]=

Quantity	Ξ_R^{Zero}	Ψ_{RI}^{Zero}	Ψ_{RII}^{Zero}	Ξ_R^{One}	Ψ_{RI}^{One}	Ψ_{RII}^{One}
Abs Bound	0.000160809	0.000757915	0.000160809	0.0020353	0.00358893	0.0020353
$\lvert x\rvert_2^2\,\hat{\Xi}$	0.00385544	0.00484829	0.00385544	0.0453281	0.048917	0.0453281

Bounds on the difference of explicit terms in dimension 30

Out[4323]=

Quantity	$\hat{\Xi}_{alpha}^{zero}(0)-\hat{\Xi}_{alpha}^{One}(0)$	$\hat{\Xi}_{alpha}^{zero}(e_1)-\hat{\Xi}_{alpha}^{One}(e_1)$	$\Sigma\Psi_{alphaI}^{Zero}-\Psi_{alphaI}^{One}$	$\Sigma\Psi_{alphaII}^{Zero}-\Psi_{alphaII}^{One}$
Bound Lower	0.000357268	0.00002647	0.000036705	0.00156173
Bound Upper	0	0	0.0000232765	0

Bounds on split of the coefficient $\Xi^{(0),\prime}$ in dimension 30

Out[4324]=

Quantity	$\hat{\Xi}'$	$\lvert x-e_I\rvert_2^2\,\hat{\Xi}'$	$\hat{\Xi}'_{alphaI}(e_I)$	$\Sigma\,\hat{\Xi}'_{alphaI}$	$\hat{\Xi}'_{RI}$	$\lvert x-e_I\rvert_2^2\,\hat{\Xi}'_{RI}$	$\lvert x\rvert_2^2\,\hat{\Xi}'$	$\hat{\Xi}'_{alphaII}(0)$	$\Sigma\,\hat{\Xi}'_{alphaII}$	$\hat{\Xi}'_{RII}$	$\lvert x\rvert_2^2\,\hat{\Xi}'_{RII}$
	0.0177534	0.000100318	0.00038824	0.0177412	0.0000149047	0.0000998735	0.000100184	0.0177345	0.000024356 7	0.0000105145	0.0000935921

In[4325]:=

Quantity	$\hat{\Xi}'$	$\|x-e_I\|_2^2\,\hat{\Xi}'$	$\hat{\Xi}'_{alphaI}(e_I)$	$\Sigma\,\hat{\Xi}'_{alph}$
	0.033359889593595905`	0.03395365883005604`	0.000021648124088105588`	0.0332430248

Bound on split of the coefficient $\Xi^{(0),I}$ in dimension 17

Out[4325]=

Quantity	$\hat{\Xi}'$	$\|x-e_I\|_2^2\,\hat{\Xi}'$	$\hat{\Xi}'_{alphaI}(e_I)$	$\Sigma\,\hat{\Xi}'_{alphaI}$	$\hat{\Xi}'_{RI}$	$\|x-e_I\|_2^2\,\hat{\Xi}'_{RI}$	$\|x\|_2^2\,\hat{\Xi}'$	$\hat{\Xi}'_{alphaII}(0)$	$\Sigma\,\hat{\Xi}'_{alphaII}$	$\hat{\Xi}'_{RII}$	$\|x\|_2^2\,\hat{\Xi}'_{RII}$
	0.033359 9	0.033953 7	0.0000216481	0.033243	0.00011228	0.000965671	0.000463875	0.0332212	0.00076049 7	0.0000857258	0.000251644

Algorithm to find good values for the constants

The follow is a semi-automate procedure to find appropriate value for the constants $\Gamma_i$and c_i.

How to use it: Initially, we guess a good value for the constant and make a first computation. Then, we deactivate the declaration of the constants at the very beginning of this document. Then we recompile the entire document multiple times (Menu Evaluate>>Evaluate notebook) and hope that the algorithm below converges to a fix-point for the parameters.

The idea to use the previously concluded bounds(+ ε) as new initial values for the Γs and constants. Then, recompile and how that we can conclude the same bounds starting from these values. As initial value we recommend to either use the values of the bounds at z_I, denote by s=i, which are independent of the values. Another reasonable choice would be to start with value that work in a slightly higher dimension.

In[4326]:=

```
Gamma1 = Ceiling[Max[boundF1[i], boundF1[o]], 10^-8] + 10^-8;
Gamma2 = Ceiling[Max[boundF2[i], boundF2[o]], 10^-8] + 10^-8;


GammaThreeClosed[1, 4] = Ceiling[Max[boundF3[1, i], boundF3[1, o]], 10^-8] + 10^-8;
GammaThreeClosed[2, 4] = Ceiling[Max[boundF3[2, i], boundF3[2, o]], 10^-8] + 10^-8;
GammaThree[1, 0] = N[Ceiling[Max[boundF3[3, i], boundF3[3, o]], 10^-8] + 10^-8];
GammaThree[1, 1] = Ceiling[Max[boundF3[4, i], boundF3[4, o]], 10^-8] + 10^-8;
GammaThree[1, 2] = Ceiling[Max[boundF3[5, i], boundF3[5, o]], 10^-8] + 10^-8;
GammaThree[1, 3] = Ceiling[Max[boundF3[6, i], boundF3[6, o]], 10^-8] + 10^-8;
GammaThree[2, 0] = Ceiling[Max[boundF3[7, i], boundF3[7, o]], 10^-8] + 10^-8;
GammaThree[2, 1] = Ceiling[Max[boundF3[8, i], boundF3[8, o]], 10^-8] + 10^-8;
GammaThree[2, 2] = Ceiling[Max[boundF3[9, i], boundF3[9, o]], 10^-8] + 10^-8;
GammaThree[2, 3] = Ceiling[Max[boundF3[10, i], boundF3[10, o]], 10^-8] + 10^-8;

cmu = mubOverMu[o] + 0.000001;
```