Symbolic Proof Claim 3.5

July 21, 2021

[1]:
```
from IPython.display import Math
def printMath(x):
    Math(latex(optK))

A.<N> = AsymptoticRing(growth_group='N^QQ * log(N)^ZZ', coefficient_ring=ZZ)
var('B', 'Q', 'K', 'S', 'I', 'n')
print("We have:")
display(Math(rf'q = N^Q, ||(g|f)|| = N^S, \beta=B \cdot N, k=K\cdot N, i=I␣
↪\cdot N'))

normgf = N^S
k=K*N
logq = log(N^Q)
logab = log(N)/(B*N)+O(N^(-1))
m = 1/2*Q*B*N + O(N/log(N))
print("Then:")
display(Math(rf'\log(q)= {latex(logq)}'))
display(Math(rf'\log(\alpha_\beta)= {latex(logab)}'))
display(Math(rf'm= {latex(m)}'))

logb(I) = logq/2 + (2*N-1-2*I*N)/2 * logab
print("So for the middle part, n-m<i<n+m-1, we have with i=I*N that:")
display(Math(rf'|b_i^*|= {latex(logb(I))}'))
```

```
We have:
```

$$q = N^Q, ||(g|f)|| = N^S, \beta = B \cdot N, k = K \cdot N, i = I \cdot N$$

```
Then:
```

$$\log(q) = Q \log(N)$$

$$\log(\alpha_\beta) = \frac{1}{B} N^{-1} \log(N) + O\left(N^{-1}\right)$$

$$m = \frac{1}{2} BQN + O\left(N \log(N)^{-1}\right)$$

```
So for the middle part, n-m<i<n+m-1, we have with i=I*N that:
```

$$|b_i^*| = \frac{1}{2}\left(Q - \frac{2(I-1)}{B}\right)\log(N) + \mathcal{O}(1)$$

[2]:
```
print("We now compute:")
display(Math(r'\sum_{i=n+k}^{n+m-1} \ln(|b_i^*|)'))
print("We have:")
display(Math(r'\ln(|b_{n+k}^*|) = '+rf'{latex(logb(1+K))}'))
print("And by definition:")
display(Math(r'\ln(|b_{n+m-1}^*|) = '+rf'{latex(logb(1+B*Q/2))}'))
logdet_last =1/2* (N+m-(N+K*N))*(1/2*(Q - 2*K/B)*log(N)+O(N^0))
print("So the sum equals:")
display(Math(r'\sum_{i=n+k}^{n+m-1} \ln(b_i^*|)=1/2 \cdot (n+m-(n+k)) \cdot␣
↪\ln(|b_{n+k}^*|)\\'+rf'= {latex(logdet_last)}'))
```

We now compute:

$$\sum_{i=n+k}^{n+m-1} \ln(|b_i^*|)$$

We have:

$$\ln(|b_{n+k}^*|) = \frac{1}{2}\left(Q - \frac{2K}{B}\right)\log(N) + \mathcal{O}(1)$$

And by definition:

$$\ln(|b_{n+m-1}^*|) = \mathcal{O}(1)$$

So the sum equals:

$$\sum_{i=n+k}^{n+m-1} \ln(b_i^*|) = 1/2 \cdot (n+m-(n+k)) \cdot \ln(|b_{n+k}^*|)$$
$$= \left(\frac{1}{8}(BQ - 2K)\left(Q - \frac{2K}{B}\right)\right) N\log(N) + O(N)$$

[3]:
```
logdet_sub = N*log(normgf)
print("We bound the volume of the sublattice by the Hadamard bound:")
display(Math(r'\ln(vol(L^{GF})) \leq N\cdot ||(g|f)||='+rf'␣
↪{latex(logdet_sub)}'))
logdet_int = logdet_sub-logdet_last
print("So by Corollary 3.4 we bound the volume of the intersection by")
display(Math(r'\ln(vol(\mathcal{L}_{\lbrack 0:n+k)})) \leq␣
↪\ln(vol(L^{GF}))-\sum_{i=n+k}^{n+m-1}␣
↪ln(b_i^*|)\leq'+rf'{latex(logdet_int)}'))
loglambda_int = log(k)/2 + logdet_int/k
print("Then Minkowski bounds the first minimum of the intersection by")
display(Math(r'\ln(\lambda_1(\mathcal{L}_{\lbrack 0:n+k)}))␣
↪\leq'+rf'{latex(loglambda_int)}'))
print("BKZ detects this short vector if (after projecting) it is smaller than:")
display(Math(r'\ln(|b_{n+k-\beta}^*|) = '+rf'{latex(logb(1+K-B))}'))
eq =  (-1/8*((B*Q - 2*K)*(Q - 2*K/B) - 8*S)/K + 1/2)==1/2*(Q + 2*(B - K)/B)
```

We bound the volume of the sublattice by the Hadamard bound:

$$\ln(vol(L^{GF})) \leq N \cdot ||(g|f)|| = SN \log(N)$$

So by Corollary 3.4 we bound the volume of the intersection by

$$\ln(vol(\mathcal{L}_{[0:n+k)})) \leq \ln(vol(L^{GF})) - \sum_{i=n+k}^{n+m-1} ln(b_i^*|) \leq \left(-\frac{1}{8}(BQ - 2K)\left(Q - \frac{2K}{B}\right) + S\right) N \log(N) + O(N)$$

Then Minkowski bounds the first minimum of the intersection by

$$\ln(\lambda_1(\mathcal{L}_{[0:n+k)})) \leq \left(-\frac{(BQ - 2K)\left(Q - \frac{2K}{B}\right) - 8S}{8K} + \frac{1}{2}\right) \log(N) + O(1)$$

BKZ detects this short vector if (after projecting) it is smaller than:

$$\ln(|b^*_{n+k-\beta}|) = \frac{1}{2}\left(Q + \frac{2(B-K)}{B}\right) \log(N) + \mathcal{O}(1)$$

[4]:
```
print("So we need to solve the following equation for B: ")
display(Math(latex(eq)))
solB = eq.solve(B)[1].right()
print("This has solution:")
display(Math(rf'B={latex(solB)}'))
print("We optimize K for a minimal B by taking the derivative to K, and finding␣
↪the roots w.r.t to K of:")
dBdK = derivative(solB, K)
display(Math(r'\frac{dB}{dK}'+rf'={latex(dBdK)}=0'))
print("This is equivalent to solving:")
display(Math(latex((K*Q^2 + K - 2*S)==sqrt(K^2*Q^2 + K^2 - 4*K*S + 4*S^2))))
eq2 = (K*Q^2 + K - 2*S)^2==K^2*Q^2 + K^2 - 4*K*S + 4*S^2
print("And by squaring both sides: ")
display(Math(latex(eq2)))
print("Which has solution: ")
display(Math(latex(eq2.solve(K)[0])))
optK = eq2.solve(K)[0].right()
assume(S>0)
print("Replacing K by this value gives the solution")
display(Math(r'B='+rf'{(latex(solB(K=optK).expand().simplify_full()))}'))
print("This concludes the proof.")
```

So we need to solve the following equation for B:

$$-\frac{(BQ - 2K)\left(Q - \frac{2K}{B}\right) - 8S}{8K} + \frac{1}{2} = \frac{1}{2}Q + \frac{B-K}{B}$$

This has solution:

$$B = -\frac{2\left(K - 2S - \sqrt{K^2Q^2 + K^2 - 4KS + 4S^2}\right)}{Q^2}$$

We optimize K for a minimal B by taking the derivative to K, and finding the roots w.r.t to K of:

$$\frac{dB}{dK} = \frac{2\left(\frac{KQ^2+K-2\,S}{\sqrt{K^2Q^2+K^2-4\,KS+4\,S^2}} - 1\right)}{Q^2} = 0$$

This is equivalent to solving:

$$KQ^2 + K - 2\,S = \sqrt{K^2Q^2 + K^2 - 4\,KS + 4\,S^2}$$

And by squaring both sides:

$$\left(KQ^2 + K - 2\,S\right)^2 = K^2Q^2 + K^2 - 4\,KS + 4\,S^2$$

Which has solution:

$$K = \frac{4\,S}{Q^2 + 1}$$

Replacing K by this value gives the solution

$$B = \frac{8\,S}{Q^2 + 1}$$

This concludes the proof.

[ ]: