PSEUDO–CODE FOR THE MLLL ALGORITHM * †

Keith Matthews

The following pseudo–code is extracted from the CALC source file lll.c (available at `http://www.numbertheory.org/calc/krm_calc.html`) for the function *BASIS_REDUCTION()*, which performs the MLLL algorithm of M. Pohst, J. Symbolic Computation (1987) 4, 123–127. We work in integers in the style of pages 329–332 of Benne de Weger's paper *Solving exponential Diophantine equations using lattice basis reduction algorithms*, J. Number Theory 26 (1987) 325–367.

*6th October 1997

†Revised and corrected 15th September 2011

In Pohst's MLLL algorithm, an integer matrix A whose rows are not necessarily LI over $\mathbb{Q}$ is reduced to a matrix A', whose first ρ rows constitute a LLL reduced matrix B and whose remaining σ rows are zero. A transformation matrix P, where $PA = A'$, is also returned. The last σ rows of P form a basis for the lattice of row vectors X such that $XA = 0$.

The Gram–Schmidt process plays an additional role to its usual one in the LLL algorithm (where its role is restricted to vectors which are LI) and is used to detect when row β is a LC of the preceding LI rows. The termination of the algorithm is guaranteed by an ingenious trick whereby the possibility that a dependency $\mathbf{b}_k = 0$ and $\mu_{k,k-1} \neq 0$ can occur only finitely many times during the course of the algorithm.

INPUT:$m \times n$ integer matrix A;
$m_1 := 1;\ n_1 := 1; D_0 := 1;\ B := A;\ P := I_n;$
$rowsB := m; K_1 := 0; \tau := 2; \sigma := 0;$
found:
if $(K_1 = 0)$
    $i := 1;$
else
    $i := K_1;$
while $(i \leq rowsB)$
    $\mathbf{c}_i := \mathbf{b}_i;$ // $\mathbf{c}_i = D_{i-1}\mathbf{b}_i^*$
    for $j = 1, \ldots, i-1$
        $\lambda_{ij} := \mathbf{b}_i \cdot \mathbf{c}_j;$ // $\lambda_{ij} = D_j \mu_{ij}$
        $\mathbf{c}_i := (D_j \mathbf{c}_i - \lambda_{ij}\mathbf{c}_j)/D_{j-1};$
    $flag := 1;$
    if $(\mathbf{c}_i \neq 0)$
        $flag := 0;$
    if $(flag = 1)$
        break;
    else
        $D_i := (c_i \cdot c_i)/D_{i-1};$ // $||\mathbf{b}_i^*||^2 = D_i/D_{i-1}$
        $i := i + 1;$
if $(flag = 1)$
    $\beta := i;$
else
    $\beta := i - 1;$
$\rho := K_1 = i - 1;$

$k := \tau$;
while $k \leq \beta$
$Flag :=$ *Reduce* $(k, k-1)$;
if $(Flag = 1)$ // Step 9 of Pohst
$\sigma := \sigma + 1$; // relation vector # σ found
$\tau := k$;
$k := k + 1$;
goto found;
if $(n_1(D_{k-2}D_k + \lambda^2_{k\,k-1}) < m_1 D^2_{k-1})$ {
$flagg := 0$;
if $(D_k = 0$ & $\lambda_{k,k-1} = 0)$
$D_{k-1} := 0$;
Swap1(k); // This changes the last two rows of B
if$(k - 1 < K_1)$
$K_1 := k - 1$;
$\mathbf{c}_{k-1} := 0$;
$\beta := \beta - 1$;
if $(k > 2)$
$k := k - 1$;
continue;
if $(flagg = 0)$
Swap2 (k);
Swap1 (k);
if $(k - 2 < K_1)$
$K_1 := k - 2$;
if $(k > 2)$
$k := k - 1$;
}

else
for $i = k-2, \ldots, 1$
$Flag :=$ *Reduce* (k, i);
if $(Flag = 1)$
$\sigma := \sigma + 1$; /* relation vector # σ found */
$\tau := k$;
$k := k + 1$;
goto found;

OUTPUT: $\rho \times n$ LLL reduced matrix B whose rows form a lattice basis for row lattice of A.
$\sigma = m - \rho$, $\mathbf{h}_{\rho+1}, \ldots, \mathbf{h}_n$ form a lattice basis for the lattice $XA = 0$.

Reduce (k, i)

```
Flag := 1;
if 2|λ_ki| > D_i
    q := ⌈λ_ki / D_i⌋;
    else q := 0;

if (q ≠ 0)
    b_k := b_k − q b_i;
    p_k := p_k − q p_i;
    λ_ki := λ_ki − q D_i;
    for j = 1, ..., i − 1
        λ_kj := λ_kj − q λ_ij;
if (b_k ≠ 0)
    Flag := 0;
if (Flag = 1)
    B := DeleteRow(k, B);
    rowsB := rowsB − 1
    for j = k, ..., m − 1
        P := SwapRows(j, j + 1, P);
return (Flag);
```


Swap1 (k)

```
b_k ↔ b_{k−1};
p_k ↔ p_{k−1};
for j = 1, ..., k − 2
    λ_kj ↔ λ_{k−1 j};
```

Swap2 (k, β)

```
for i = k + 1, ..., β {
    t := λ_{i k−1} D_k − λ_ik λ_{k k−1};
    λ_{i k−1} := (λ_{i k−1} λ_{k k−1} + λ_ik D_{k−2}) / D_{k−1};
    λ_ik := t / D_{k−1};
}
D_{k−1} := (D_{k−2} D_k + λ_{k k−1}^2) / D_{k−1};
```

Remarks.

1. K_1 is the number of LI rows of B found after G–S process.

2. $flag = 0$ means the $\rho = \beta$ rows of B are LI.

3. $flag = 1$ means the first $\rho = \beta - 1$ rows of B are LI, but row β is a LC of the preceding rows.

4. β is the number of rows of B currently being examined.