Note di Matematica **22,** n. 1, 2003, 27–48.

Five constructions of representations of quantum groups

Willem A. de Graaf
University of Utrecht, The Netherlands
degraaf@math.uu.nl

Received: 7/11/2002; accepted: 8/11/2002.

Abstract. We describe a number of constructions of irreducible representations of quantized enveloping algebras of semisimple Lie algebras. In the final section the methods are compared in some practical examples.

Keywords: Representation Theory, Quantum Groups, Algorithms.

MSC 2000 classification: 17B37, 68W30.

Introduction

Let $U_q(\mathfrak{g})$ be the quantized enveloping algebra of the semisimple Lie algebra $\mathfrak{g}$. The irreducible representations of $U_q(\mathfrak{g})$ are parametrized by the dominant weights in the weight lattice. A lot is known about these irreducible representations as they have the same dimensions and weight systems as the irreducible representations of $\mathfrak{g}$. So by Weyl's dimension formula we can compute the dimension, and by Freudenthal's formula we can compute the weight-multiplicities. Also in this case we have the additional combinatorial tool of the crystal graph.

These combinatorial methods provide an easy way of obtaining information about a particular irreducible representation. However, they do not say much about how an element of $U_q(\mathfrak{g})$ acts, e.g., what its matrix is with respect to a basis of the underlying module. This paper oulines a few methods to tackle this last problem. We will describe algorithms to construct a basis of the irreducible $U_q(\mathfrak{g})$-module with a given highest weight λ, along with methods to compute the action of an element of $U_q(\mathfrak{g})$.

In Section 1 we describe the theoretical set up and the notation that we are using. Then in Sections 2, 3, 4, 5, 6 we describe five methods for constructing an irreducible representation of $U_q(\mathfrak{g})$. The main ingredient of theses sections is respectively, Gröbner bases, canonical bases, tensor products, dual spaces, and Gelfand-Zetlin patterns. Not all of these methods are equally original. In particular, the algorithm using Gelfand-Zetlin patterns is taken straight from the literature. In the final section we compare the methods in some practical examples.

1 Preliminaries

Let $\mathfrak{g}$ be a semisimple Lie algebra over $\mathbb{C}$. By Φ we denote the root system of $\mathfrak{g}$, and $\Delta = \{\alpha_1, \ldots, \alpha_l\}$ will be a fixed set of simple roots of Φ. Let W denote the Weyl group of Φ, which is generated by the simple reflections $s_i = s_{\alpha_i}$ for $1 \leq i \leq l$. Let $\mathbb{R}\Phi$ be the vector space over $\mathbb{R}$ spanned by Φ. On $\mathbb{R}\Phi$ we fix a W-invariant inner product $(\ ,\ )$ such that $(\alpha, \alpha) = 2$ for short roots α. This means that $(\alpha, \alpha) = 2, 4, 6$ for $\alpha \in \Phi$.

We work over the field $\mathbb{Q}(q)$. For $\alpha \in \Phi$ set $q_\alpha = q^{(\alpha,\alpha)/2}$. For $n \in \mathbb{Z}$ we set $[n]_\alpha = q_\alpha^{-n+1} + q_\alpha^{-n+3} + \cdots + q_\alpha^{n-1}$. Also $[n]_\alpha! = [n]_\alpha [n-1]_\alpha \cdots [1]_\alpha$ and

$$\begin{bmatrix} n \\ k \end{bmatrix}_\alpha = \frac{[n]_\alpha!}{[k]_\alpha! [n-k]_\alpha!}.$$

Let $\Delta = \{\alpha_1, \ldots, \alpha_l\}$ be a simple system of Φ. Then the quantized enveloping algebra $U_q = U_q(\mathfrak{g})$ is the associative algebra (with one) over $\mathbb{Q}(q)$ generated by F_α, K_α, K_α^{-1}, E_α for $\alpha \in \Delta$, subject to the following relations

$$\begin{aligned}
K_\alpha K_\alpha^{-1} &= K_\alpha^{-1} K_\alpha = 1,\ K_\alpha K_\beta = K_\beta K_\alpha \\
E_\beta K_\alpha &= q^{-(\alpha,\beta)} K_\alpha E_\beta \\
K_\alpha F_\beta &= q^{-(\alpha,\beta)} F_\beta K_\alpha \\
E_\alpha F_\beta &= F_\beta E_\alpha + \delta_{\alpha,\beta} \frac{K_\alpha - K_\alpha^{-1}}{q_\alpha - q_\alpha^{-1}} \\
&\sum_{k=0}^{1-\langle \beta, \alpha^\vee \rangle} (-1)^k \begin{bmatrix} 1 - \langle \beta, \alpha^\vee \rangle \\ k \end{bmatrix}_\alpha E_\alpha^{1-\langle \beta, \alpha^\vee \rangle - k} E_\beta E_\alpha^k = 0 \\
&\sum_{k=0}^{1-\langle \beta, \alpha^\vee \rangle} (-1)^k \begin{bmatrix} 1 - \langle \beta, \alpha^\vee \rangle \\ k \end{bmatrix}_\alpha F_\alpha^{1-\langle \beta, \alpha^\vee \rangle - k} F_\beta F_\alpha^k = 0,
\end{aligned}$$

where the last two relations are for all $\alpha \neq \beta$.

Let U^-, U^0, U^+ be the subalgebras of U_q generated by respectively, F_α for $\alpha \in \Delta$, $K_\alpha^{\pm 1}$ for $\alpha \in \Delta$, and E_α for $\alpha \in \Delta$. Then as a vector space $U_q \cong U^- \otimes U^0 \otimes U^+$ ([9], Theorem 4.21).

We describe the bases of U^-, U^+ that we use. For $\alpha \in \Delta$ we have an automorphism $T_\alpha : U_q \to U_q$, which is determined by the formulas in [9], 8.14. Let w_0 denote the longest element of W. Then by $R(w_0)$ we denote the set of sequences $\underline{i} = (i_1, \ldots, i_t)$ such that $s_{i_1} \cdots s_{i_t}$ is a reduced expression for w_0. Let $\underline{i} \in R(w_0)$ and set $F_k = T_{\alpha_{i_1}} \cdots T_{\alpha_{i_{k-1}}}(F_{\alpha_{i_k}})$ and $E_k = T_{\alpha_{i_1}} \cdots T_{\alpha_{i_{k-1}}}(E_{\alpha_{i_k}})$ for $1 \leq k \leq t$. Set $F_k^{(n)} = F_k^n / [n]_{\beta_k}!$, where $\beta_k = s_{i_1} \cdots s_{i_{k-1}}(\alpha_{i_k})$, and similarly for $E_k^{(n)}$. Then the set $B_{\underline{i}}^- = \{F_1^{(n_1)} \cdots F_t^{(n_t)} \mid n_i \geq 0\}$ forms a basis of U^-. Likewise $B_{\underline{i}}^+ = \{E_1^{(n_1)} \cdots E_t^{(n_t)} \mid n_i \geq 0\}$ forms a basis of U^+. Now using the isomorphism $U_q \cong U^- \otimes U^0 \otimes U^+$ we get a basis of U_q, called a PBW-type basis. In [5] an algorithm is described for expressing the product of two elements of a PBW-type basis as a linear combination of basis elements.

Let $\nu = \sum_{i=1}^l k_i\alpha_i$, where the k_i are non-negative integers. Then we let U_ν^- be the subspace of U^- spanned by all products $F_{\alpha_{i_1}} \cdots F_{\alpha_{i_r}}$, such that α_i appears k_i times. The elements of U_ν^- are said to be of weight ν, and we write $\mathrm{wt}(a) = \nu$ for $a \in U_\nu^-$. Then $\mathrm{wt}(F_1^{(n_1)} \cdots F_t^{(n_t)}) = \sum m_k\beta_k$. We remark that in [9], elements of U_ν^- are said to be of degree $-\nu$.

Let P denote the weight lattice of Φ, and let P^+ be the set of dominant weights. Let V be an irreducible U_q-module. Then there is a $\lambda \in P^+$, and a $v_\lambda \in V$ (unique upto scalar multiples) such that $E_\alpha \cdot v_\lambda = 0$, $K_\alpha^{\pm 1} \cdot v_\lambda = q^{\pm(\alpha,\lambda)} v_\lambda$, and $V = U_q^- \cdot v_\lambda$. The λ is called the highest weight of V, and v_λ a highest-weight vector. On the other hand, given a $\lambda \in P^+$, an irreducible module with highest weight λ is constructed as follows. Let $J(\lambda)$ be the left ideal of U_q generated by E_α, $K_\alpha^{\pm 1} - q^{\pm(\alpha,\lambda)}$. Then $M(\lambda) = U_q/J(\lambda)$ is a U_q-module, called a Verma module. We have $U_q = U^- \oplus J(\lambda)$ so $M(\lambda) \cong U^-$ as a U^--module. Write $\lambda = r_1\lambda_1 + \cdots + r_l\lambda_l$, where the λ_i are the fundamental weights. Let $I(\lambda)$ be the U_q-submodule of $M(\lambda)$ generated by $F_{\alpha_i}^{r_i+1}$ for $1 \leq i \leq l$. Then $V(\lambda) = M(\lambda)/I(\lambda)$ is an irreducible U_q-module with highest weight λ (cf. [9], Theorem 5.10). Let v_λ denote the image of $1 \in U_q$ in $V(\lambda)$. Then v_λ is a highest-weight vector. Finally we remark that $I(\lambda)$ is equal to the left-ideal of U^- generated by the $F_{\alpha_i}^{r_i+1}$ (under the isomorphism $U^- \cong M(\lambda)$).

We have that $V(\lambda)$ is the direct sum of weight spaces $V(\lambda)_\mu$, where $\mu \in P$, and $V(\lambda)_\mu = \{v \in V(\lambda) \mid K_\alpha \cdot v = q^{(\mu,\alpha)} v\}$. The elements of $V(\lambda)_\mu$ are called weight vectors of weight μ. We let $P(\lambda)$ be the set of all μ such that $\dim V(\lambda)_\mu > 0$. We have algorithms for computing the elements of $P(\lambda)$, and $\dim V(\lambda)_\mu$ for $\mu \in P(\lambda)$.

Let $\mathbb{R}P$ be the vector space over $\mathbb{R}$ spanned by P, i.e., $\mathbb{R}P = \mathbb{R}\lambda_1 \oplus \cdots \oplus \mathbb{R}\lambda_l$. Let Π be the set of piecewise linear paths $\pi : [0,1] \to \mathbb{R}P$, such that $\pi(0) = 0$. For $\alpha \in \Delta$ Littelmann defined operators $e_\alpha, f_\alpha : \Pi \to \Pi \cup \{0\}$ (cf. [14], [15]), with the following property. Let $\lambda \in P^+$ be a dominant weight, and let π_λ be the path given by $\pi_\lambda(t) = \lambda t$ (i.e., a straight line from the origin to λ). let Π_λ be the set of all $f_{\alpha_{i_1}} \cdots f_{\alpha_{i_k}}(\pi_\lambda)$. Then all paths in Π_λ end in an element of P. Furthermore, the number of paths ending in $\mu \in P$ is equal to $\dim V(\lambda)_\mu$.

The action of the path operators can be encoded in a directed labeled graph Γ_λ. The points of Γ_λ are the paths in Π_λ, and there is an edge $\pi_1 \xrightarrow{\alpha} \pi_2$ if $f_\alpha(\pi_1) = \pi_2$. This graph is isomorphic to the crystal graph of $V(\lambda)$ ([11]). This fact will be used frequently throughout the paper.

2 Using Gröbner bases

In this section we describe an algorithm for computing a Gröbner basis of the left ideal $I(\lambda)$. This will give us a method for constructing $V(\lambda)$.

We fix an $\underline{i} \in R(w_0)$, and let $B^-_{\underline{i}}$ be the corresponding basis of U^-, consisting of monomials $x = F_1^{(n_1)} \cdots F_t^{(n_t)}$. In the sequel we will refer to the n_i as the exponents of x.

The lexicographical order on sequences of length t is defined as follows: $(m_1, \dots, m_t) <_{\mathrm{lex}} (n_1, \dots, n_t)$ if there is a $k \geq 1$ such that $m_1 = n_1, \cdots, m_{k-1} = n_{k-1}$ and $m_k < n_k$. Let $x_1, x_2 \in B^-_{\underline{i}}$ with exponents m_i and n_i respectively. Then we write $x_1 <_{\mathrm{lex}} x_2$ if $(m_i) <_{\mathrm{lex}} (n_i)$. Let $f \in U^-$, then by $\mathrm{LM}(f)$ we denote the biggest monomial of $B^-_{\underline{i}}$ appearing in f (in the order $<_{\mathrm{lex}}$).

1 Lemma. *Let $x_1, x_2 \in B^-_{\underline{i}}$ with exponents m_i and n_i respectively. Then* $\mathrm{LM}(x_1x_2) = F_1^{(m_1+n_1)} \cdots F_t^{(m_t+n_t)}$.

PROOF. First we prove the result for the case where $x_1 = F_k$, by induction on $(n_1, \dots, n_t)$, where these tuples are ordered lexicographically. Let j be the smallest index such that $n_j > 0$. If $k \leq j$ then there is nothing to prove, so suppose that $k > j$. Then $F_kF_j = q^{-(\beta_k,\beta_j)}F_kF_j + \sum_r \xi_r y_r$, where $y_r \in B^-_{\underline{i}}$ are monomials only involving $F_{j+1}, \dots, F_{k-1}$, (see [2], [5]). Hence

$$F_kF_jF_j^{(n_j-1)} \cdots F_t^{(n_t)} = \zeta F_jF_kF_j^{(n_j-1)} \cdots F_t^{(n_t)} + \sum_r \xi_r y_r F_j^{(n_j-1)} \cdots F_t^{(n_t)}.$$

By induction the leading monomial of the first term is $F_j^{(n_j)} \cdots F_k^{(n_k+1)} \cdots F_t^{(n_t)}$. Since the y_r only involve $F_{j+1}, \dots, F_{k-1}$, when rewriting $y_rF_j^{(n_j-1)} \cdots F_t^{(n_t)}$ to a linear combination of elements of $B^-_{\underline{i}}$ we will never produce an extra F_j. Hence

$$y_rF_j^{(n_j-1)} \cdots F_t^{(n_t)} = \sum_s \eta_s F_j^{(p_j^s)} \cdots F_t^{(p_t^s)},$$

where $p_j^s \leq n_j - 1$. So we get the lemma in this case. The general case is done by induction on $(m_1, \dots, m_t)$. $\boxed{QED}$

Now let $x, y \in B^-_{\underline{i}}$. Then x is said to be a *factor* of y if there is an $x' \in B^-_{\underline{i}}$ with $\mathrm{LM}(x'x) = y$. Note that Lemma 1 gives an easy method for deciding whether x is a factor of y, and for finding the x'.

Let $f \in U^-$, and $G \subset U^-$ a finite set. Then we have the following algorithm for left-reducing f modulo G. Initially we set $h = f$, and $r = 0$. Let $x = \mathrm{LM}(h)$, with coefficient c. Suppose that there is a g in G such that $\mathrm{LM}(g)$ is a factor of x. Then we set $h := h - c'yg$, where $y \in B^-_{\underline{i}}$ is such that $y\mathrm{LM}(g)$ has leading monomial x, and the coefficient c' is chosen such that the leading monomials cancel. If there is no such g, then we set $h := h - cx$ and $r := r + cx$, and continue. The process stops when $h = 0$. This algorithm terminates because $\mathrm{LM}(h)$ decreases every round, and there are no infinite descending sequences of monomials in $B^-_{\underline{i}}$. Furthermore, upon termination we have $f = r \bmod I$, where I is the left-ideal of U^- generated by G. Also no $\mathrm{LM}(g)$ for $g \in G$ is a factor of any monomial occurring in r. The element r is called the *left-remainder* of f modulo G.

A set $G \subset U^-$ is called a *Gröbner basis* of the left-ideal $I \subset U^-$ if for all $f \in I$ there is a $g \in G$ such that $\mathrm{LM}(g)$ is a factor of $\mathrm{LM}(f)$. We remark that this is equivalent to saying that all $f \in I$ have left-remainder 0 modulo G. Also, if G is a Gröbner basis of I then the cosets of all elements $x \in B^-_{\underline{i}}$ that have no factors $\mathrm{LM}(g)$ for $g \in G$, form a basis of U^-/I.

Let $f, g \in U^-$, and write $\mathrm{LM}(f) = F_1^{(m_1)} \cdots F_t^{(m_t)}$, $\mathrm{LM}(g) = F_1^{(n_1)} \cdots F_t^{(n_t)}$. Set $k_i = \max(m_i - n_i, 0)$ and $l_i = \max(n_i - m_i, 0)$, and $x = F_1^{(k_1)} \cdots F_t^{(k_t)}$, $y = F_1^{(l_1)} \cdots F_t^{(l_t)}$. Let c_1 be the coefficient of $\mathrm{LM}(yf)$ (in yf), and c_2 the coefficient of $\mathrm{LM}(xg)$ (in xg). Then $S(f,g) = c_2 yf - c_1 xg$ is called the S-element of f and g. The proof of the next lemma is analogous to the proof of the same result for universal enveloping algebras of Lie algebras (cf. [4]). Therefore we omit it.

2 Lemma. *Let $G \subset U^-$, and suppose that the left-remainder of $S(g_1, g_2)$ modulo G is 0 for all $g_1, g_2 \in G$. Then G is a Gröbner basis for the left-ideal generated by G.*

This means that we have the following procedure for computing a Gröbner basis of a (finitely-generated) left-ideal I. Initially G will be equal to a finite generating set of I. We make S-elements of $g_1, g_2 \in G$ and add their left-remainders modulo G to G, until the left-remainders of all possible S-elements are zero. This terminates because otherwise it is possible to construct an infinite increasing series of ideals in the polynomial ring of t variables. The algorithm returns a Gröbner basis by Lemma 2.

However, for the case where $I = I(\lambda)$ we know some additional data, which we can use. Let $x \in B^-_{\underline{i}}$. Then the weight of $x \cdot v_\lambda$ in $V(\lambda)$ is $\lambda - \mathrm{wt}(x)$. In order to compute a Gröbner basis of $I(\lambda)$ we first compute the set $P(\lambda)$, and $\dim V(\lambda)_\mu$ for all $\mu \in P(\lambda)$. Initially we let G be the set of all $x \in B^-_{\underline{i}}$ such that $\lambda - \mathrm{wt}(x) \notin P(\lambda)$. This is an infinite set, but we will only be using a finite number of elements of it, so this poses no problems. Also we set $M = \emptyset$. All weights $\mu \in P(\lambda)$ are of the form $\lambda - \sum_i k_i \alpha_i$, where the k_i are non-negative integers. We say that $\sum_i k_i$ is the *level* of μ. Then we loop through $P(\lambda)$, according to increasing level. For $\mu \in P(\lambda)$ we do the following:

(1) Let M_μ be the set of all $x \in B^-_{\underline{i}}$ such that $\mathrm{wt}(x) = \lambda - \mu$ and there is no $g \in G$ such that $\mathrm{LM}(g)$ is a factor of x.

(2) As long as $|M_\mu| > \dim V(\lambda)_\mu$ we do the following. Make an S-element (of weight $\lambda-\mu$) of $g_1, g_2 \in G$, and let h be its left-remainder modulo G. If $h \neq 0$, then add h to G and erase its leading monomial from M_μ.

(3) Add the elements of M_μ to M.

Step 2. terminates because the straightforward Gröbner basis algorithm terminates. Upon termination G is a Gröbner basis and the cosets of the elements of M modulo $I(\lambda)$ form a basis of $U^-/I(\lambda)$.

This is much more efficient than just doing the straightforward Gröbner basis algorithm, because far fewer S-elements are checked. The stopping criterion is now whether we have the right dimension, instead of whether all S-elements reduce to 0.

Finally we note that a Gröbner basis G of $I(\lambda)$ gives us a method for computing $u \cdot x$, where $u \in U_q$, $x \in M$. First we compute $u \cdot x$ in $M(\lambda)$ which gives us an element $f \in U^-$. Then we calculate the left remainder of f modulo G. This will be the linear combination of elements of M that we are looking for.

3 Example. We let Φ be the root system of type A_2, with simple roots α, β. We use the reduced expression $s_\alpha s_\beta s_\alpha$ for w_0. Then $F_1 = F_\alpha$, $F_2 = T_\alpha(F_\beta)$, and $F_3 = T_\alpha T_\beta(F_\alpha) = F_\beta$. Denoting F_2 by $F_{\alpha+\beta}$, we have

$$\begin{aligned} F_{\alpha+\beta}F_\alpha &= q^{-1}F_\alpha F_{\alpha+\beta} \\ F_\beta F_\alpha &= qF_\alpha F_\beta + F_{\alpha+\beta} \\ F_\beta F_{\alpha+\beta} &= q^{-1}F_{\alpha+\beta}F_\alpha. \end{aligned}$$

In the sequel (m,n) will denote the weight $m\lambda_1 + n\lambda_2$. We use the algorithm described in this section to construct $V(\lambda)$, where $\lambda = (1,1)$. Figure 1 displays the weights of $V(\lambda)$. In this picture, the highest weight is on top, and weights of the same level are on the same line. If a weight ν occurs to the left of a weight μ appearing on the line above, then $\nu = \mu - \alpha$, if it occurs on the right, then $\nu = \mu - \beta$. Also the superscript indicates the dimension of the corresponding weight space.

We go through the weights, from top to bottom. First of all, v_λ, $F_\alpha v_\lambda$, $F_\beta v_\lambda$ are basis vectors of weights $(1,1)$, $(-1,2)$ and $(2,-1)$ respectively. Also there are two possible basis vectors of weight $(0,0)$: $F_\alpha F_\beta v_\lambda$ and $F_{\alpha+\beta}v_\lambda$. However, since the dimension of the corresponding weight space is 2, these must be linearly independent. Now we consider the weight $\mu = (-2,1)$. Here $M_\mu = \{F_\alpha F_{\alpha+\beta}\}$, as the other possible element, $F_\alpha^{(2)}F_\beta$ is excluded because $F_\alpha^{(2)}F_\beta$ has $F_\alpha^{(2)} \in I(\lambda)$ as a factor. So in this case we do nothing in Step 2 of the algorithm, and we have found all basis elements of weight μ. Similarly, the only basis element of weight $(1,-2)$ is $F_{\alpha+\beta}F_\beta v_\lambda$. Now we look at the weight $\mu = (-1,-1)$. We

$$
\begin{array}{ccc}
 & (1,1)^1 & \\
(-1,2)^1 & & (2,-1)^1 \\
 & (0,0)^2 & \\
(-2,1)^1 & & (1,-2)^1 \\
 & (-1,-1)^1 &
\end{array}
$$

Figure 1. Weights of $V(1,1)$.

have $M_\mu = \{F_\alpha F_{\alpha+\beta} F_\beta, F_{\alpha+\beta}^{(2)}\}$. However, $\dim V(\lambda)_\mu = 1$ so here we have to do some work. We compute the S-element $S(F_\alpha^{(2)}, F_\beta^{(2)}) = q^4 F_\alpha^{(2)} F_\beta^{(2)} - F_\beta^{(2)} F_\alpha^{(2)} = -qF_\alpha F_{\alpha+\beta} F_\beta - F_{\alpha+\beta}^{(2)}$. From this we see that $F_\alpha F_{\alpha+\beta} F_\beta v_\lambda = -q^{-1} F_{\alpha+\beta}^{(2)} v_\lambda$, and the basis element of weight μ that remains is $F_{\alpha+\beta}^{(2)} v_\lambda$. So we have found a basis of $V(\lambda)$, and we can express every element $u \cdot v_\lambda$ (for $u \in U^-$) as a linear combination of basis elements.

4 Remark. The algorithm for computing irreducible representations of semisimple Lie algebras of [6] is highly similar to the algorithm described in this section.

5 Remark. It is also possible to use the reverse lexicographical order, instead of the lexicographical order. However, degree compatible orders do not work here as the product $x_1 x_2$ may contain monomials of degree larger than the sum of the degrees of x_1 and x_2.

3 Using canonical bases

We let $^-$ be the automorphism of U^- given by $\overline{F}_\alpha = F_\alpha$, and $\overline{q} = q^{-1}$ (see [9], Proposition 11.9). Then by results of Kashiwara and Lusztig there is a unique basis $\mathbf{B}$ of U^-, called the canonical basis, such that for $b \in \mathbf{B}$ we have

(1) $\overline{b} = b$,

(2) for any $\underline{i} \in R(w_0)$, $b = x + \sum_j \zeta_j x_j$, where $x, x_j \in B_{\underline{i}}^-$ are all of the same weight, and $\zeta_j \in q\mathbb{Z}[q]$

(cf. [18], Proposition 8.2). If we use a fixed $\underline{i} \in R(w_0)$ and write $b \in \mathbf{B}$ as $b = x + \sum_j \zeta_j x_j$ with $x, x_j \in B_{\underline{i}}^-$, then we refer to x as the *principal monomial* of b. For every $x \in B_{\underline{i}}^-$ there is a unique $b \in \mathbf{B}$ with principal monomial equal to x, and we write $b = G_{\underline{i}}(x)$.

The importance of $\mathbf{B}$ for constructing $V(\lambda)$ lies in the following fact. For any dominant $\lambda \in P^+$ we have $\mathbf{B} = \mathbf{B}_\lambda \cup \mathbf{B}^\lambda$ (disjoint union), where $\mathbf{B}^\lambda$ is a basis of $I(\lambda)$ ([9], Theorem 11.10). However, in order to use this we must solve two problems: first of all we must have a method for finding elements of $\mathbf{B}$, and secondly we need a way to decide whether a given element of $\mathbf{B}$ lies in $\mathbf{B}_\lambda$.

First we describe an algorithm for computing $G_{\underline{i}}(x)$, that is different from the one contained in [8]. We fix $\underline{i} \in R(w_0)$. In [8] the following was proved:

$$G_{\underline{i}}(x) = x + \sum_{x_j >_{\mathrm{lex}} x} \zeta_j x_j, \tag{1}$$

for certain $x, x_j \in B_{\underline{i}}^-$, and $\zeta_j \in q\mathbb{Z}[q]$. This means that we also have

$$G_{\underline{i}}(x) = x + \sum_{x_j >_{\mathrm{lex}} x} \delta_j G(x_j),$$

for certain $\delta_j \in q\mathbb{Z}[q]$. After taking images under $^-$ and subtracting we get:

$$\overline{x} - x = \sum_{x_j >_{\mathrm{lex}} x} (\delta_j - \overline{\delta}_j) G(x_j).$$

Let $x \in B_{\underline{i}}^-$. If there is no $y \in B_{\underline{i}}^-$ of the same weight as x, with $y >_{\mathrm{lex}} x$, then $G_{\underline{i}}(x) = x$. Otherwise we assume that we have computed $G_{\underline{i}}(y)$ for $y >_{\mathrm{lex}} x$ and perform the following steps to compute $G_{\underline{i}}(x)$.

(1) Write $\overline{x} - x$ as $\sum_{x_j >_{\mathrm{lex}} x} d_j G_{\underline{i}}(x_j)$.

(2) Let δ_j be the unique element of $q\mathbb{Z}[q]$ such that $\delta_j - \overline{\delta}_j = d_j$. Return $x + \sum_{x_j >_{\mathrm{lex}} x} \delta_j G_{\underline{i}}(x_j)$.

6 Remark. Experiments suggest that this method is more efficient than the one described in [8], when the rank of the root system is 2, 3. For ranks 4, 5 the methods perform about equal, wheras for ranks ≥ 6 the algorithm from [8] is more efficient.

Now we deal with our second problem: deciding whether a given element of $\mathbf{B}$ lies in $\mathbf{B}_\lambda$ or in $\mathbf{B}^\lambda$. For $\alpha \in \Delta$ the Kashiwara operator $\widetilde{F}_\alpha : B_{\underline{i}}^- \to B_{\underline{i}}^-$ is defined as follows. Let $\underline{i}, \underline{i}'$ be two elements from $R(w_0)$. Let $x \in B_{\underline{i}}^-$ and write $b = G_{\underline{i}}(x)$. Then there is an $x' \in B_{\underline{i}'}$ with $b = G_{\underline{i}'}(x')$. Set $R_{\underline{i}}^{\underline{i}'}(x) = x'$. Then $R_{\underline{i}}^{\underline{i}'} : B_{\underline{i}}^- \to B_{\underline{i}'}^-$ (compare [17] 42.1.3). Now let $\underline{i}' = (i_1', \ldots, i_t') \in R(w_0)$ be such that $\alpha_{i_1'} = \alpha$. Let $x' = R_{\underline{i}}^{\underline{i}'}(x)$ and let x'' be the element of $B_{\underline{i}'}$ obtained from x' by increasing its first exponent by 1. Set $\widetilde{F}_\alpha(x) = R_{\underline{i}'}^{\underline{i}}(x'')$. We refer to [8] for an algorithm for computing $\widetilde{F}_\alpha$ (without constructing $\mathbf{B}$).

Now let Γ_λ be the crystal graph of $V(\lambda)$ (see Section 1). For $\pi \in \Pi_\lambda$ we fix a sequence of simple roots $\eta_\pi = (\alpha_{i_1}, \ldots, \alpha_{i_r})$ such that $\pi = f_{\alpha_{i_1}} \cdots f_{\alpha_{i_r}} \cdot \pi_\lambda$. Set

$$x_\pi = \widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_r}}(1).$$

7 Lemma. *We have that x_π does not depend on the choice of η_π. Furthermore, $G_{\underline{i}}(x) \in \mathbf{B}_\lambda$ if and only if $x = x_\pi$ for some $\pi \in \Pi_\lambda$. Also, the set $\{x_\pi \cdot v_\lambda \mid \pi \in \Pi_\lambda\}$ is a basis of $V(\lambda)$.*

Proof. For the proof we borrow some notation and results from [9]. Let $(\mathcal{L}(\lambda), \mathcal{B}(\lambda))$ be the crystal base of $V(\lambda)$ ([9], Chapter 9). We use the Kashiwara operators $\widetilde{F}_\alpha : \mathcal{B}(\lambda) \to \mathcal{B}(\lambda) \cup \{0\}$ for $\alpha \in \Delta$ as defined in [9], 9.2. (Note that we use the same symbol to denote the Kashiwara operator on $B^-_{\underline{i}}$; however, it will be clear which operator we mean.)

Let $\mathcal{L}(\infty)$, $\mathcal{B}(\infty)$ be as in [9], Chapter 10. Then $\mathcal{L}(\infty)$ is spanned by $G_{\underline{i}}(x)$ for $x \in B^-_{\underline{i}}$. By (1) we see that $\mathcal{L}(\infty)$ is also spanned by the elements of $B^-_{\underline{i}}$. Furthermore, $\mathcal{B}(\infty)$ consists of the cosets $x \bmod q\mathcal{L}(\infty)$ for $x \in B^-_{\underline{i}}$. So there are the Kashiwara operators $\widetilde{F}_\alpha : \mathcal{B}(\infty) \to \mathcal{B}(\infty)$ defined by $\widetilde{F}_\alpha(x \bmod q\mathcal{L}(\infty)) = \widetilde{F}_\alpha x \bmod q\mathcal{L}(\infty)$.

There is a map $\varphi_\lambda : U^- \to V(\lambda)$ defined by $\varphi_\lambda(u) = uv_\lambda$ ([9], 10.3). This induces a map $\overline{\varphi}_\lambda : \mathcal{L}(\infty)/q\mathcal{L}(\infty) \to \mathcal{L}(\lambda)/q\mathcal{L}(\lambda)$.

Now choose two sequences $\eta = (\alpha_{i_1}, \ldots, \alpha_{i_k})$, and $\eta' = (\alpha_{j_1}, \ldots, \alpha_{j_m})$ such that $f_{\alpha_{i_1}} \cdots f_{\alpha_{i_k}} \cdot \pi_\lambda = f_{\alpha_{j_1}} \cdots f_{\alpha_{j_m}} \cdot \pi_\lambda \neq 0$. Then by [11] we see that

$$\widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_k}} v_\lambda = \widetilde{F}_{\alpha_{j_1}} \cdots \widetilde{F}_{\alpha_{j_m}} v_\lambda \bmod q\mathcal{L}(\lambda).$$

Now $\overline{\varphi}_\lambda$ is a bijection between the set $\{b \in \mathcal{B}(\infty) \mid \overline{\varphi}_\lambda(b) \neq 0\}$ and $\mathcal{B}(\lambda)$ ([9], 10.14). Furthermore,

$$\overline{\varphi}_\lambda(\widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_k}}(1) \bmod q\mathcal{L}(\infty)) = \widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_k}} v_\lambda \bmod q\mathcal{L}(\lambda) \neq 0$$

([9], 10.9). Since also $\overline{\varphi}_\lambda(\widetilde{F}_{\alpha_{j_1}} \cdots \widetilde{F}_{\alpha_{j_m}}(1) \bmod q\mathcal{L}(\infty)) = \widetilde{F}_{\alpha_{j_1}} \cdots \widetilde{F}_{\alpha_{j_m}} v_\lambda \bmod q\mathcal{L}(\lambda)$, we conclude that $\widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_k}}(1) = \widetilde{F}_{\alpha_{j_1}} \cdots \widetilde{F}_{\alpha_{j_m}}(1)$.

Now suppose that $x = x_\pi$ for some $\pi \in \Pi_\lambda$, then $x = \widetilde{F}_{\alpha_{i_1}} \cdots \widetilde{F}_{\alpha_{i_k}}(1)$, where $f_{\alpha_{i_1}} \cdots f_{\alpha_{i_k}} \cdot \pi_\lambda = \pi \neq 0$. It follows that $\overline{\varphi}_\lambda(x \bmod q\mathcal{L}(\infty)) \neq 0$. Therefore $G_{\underline{i}}(x) \in \mathbf{B}_\lambda$ (cf. [9], Theorem 11.10). Now the number of monomials x_π is exactly the size of $\mathbf{B}_\lambda$. Hence every element of $\mathbf{B}_\lambda$ is of the form $G_{\underline{i}}(x_\pi)$ for $\pi \in \Pi_\lambda$. In other words, the set $\{G_{\underline{i}}(x_\pi) \cdot v_\lambda \mid \pi \in \Pi_\lambda\}$ is a basis of $V(\lambda)$. Now by using (1) and induction from above (relative to the lexicographical order) we see that every $G_{\underline{i}}(x_\pi) \cdot v_\lambda$ is a linear combination of $x_{\pi'} \cdot v_\lambda$. Therefore, $\{x_\pi \cdot v_\lambda \mid \pi \in \Pi_\lambda\}$ is also a basis of $V(\lambda)$. $\boxed{QED}$

$$
\begin{tikzcd}
 & 1 \arrow[dl, "\alpha"'] \arrow[dr, "\beta"] & \\
F_\alpha \arrow[d, "\beta"] & & F_\beta \arrow[d, "\alpha"] \\
F_{\alpha+\beta} \arrow[d, "\beta"] & & F_\alpha F_\beta \arrow[d, "\alpha"] \\
F_{\alpha+\beta}F_\beta \arrow[dr, "\alpha"'] & & F_\alpha^{(2)}F_\beta \arrow[dl, "\beta"] \\
 & F_\alpha F_{\alpha+\beta} F_\beta &
\end{tikzcd}
$$

Figure 2. Crystal graph Γ_λ, where $\lambda = \lambda_1 + \lambda_2$.

Set $M = \{x_\pi \mid \pi \in \Pi_\lambda\}$. For $x \in M$ we consider the problem of rewriting $a \cdot (x \cdot v_\lambda)$, where $a \in U_q$, as a linear combination of elements $x_\pi \cdot v_\lambda$. First we compute a $u \in U^-$ such that $a \cdot (x \cdot 1) = u \cdot 1 \in M(\lambda)$. Without loss of generality we may suppose that u is homogeneous of weight ν. Set $\mu = \lambda - \nu$. Then if μ is not a weight of $V(\lambda)$ we have $a \cdot (x \cdot v_\lambda) = 0$. Otherwise, suppose that u contains the monomial $y \in B^-_{\underline{i}}$, with $y \notin M$. Then we compute an element $G'(y)$ such that $G'(y) = y + \sum_{y' >_{\text{lex}} y} \zeta_{y',y} y'$ and $G'(y) \cdot v_\lambda = G_{\underline{i}}(y) \cdot v_\lambda = 0$. Subsequently we use $G'(y)$ to rewrite $y \cdot v_\lambda$ to a linear combination of elements $y' \cdot v_\lambda$, where $y' >_{\text{lex}} y$. Continuing this process we reach a linear combination of elements of M.

We explain how to compute $G'(y)$. Let M_ν be the set of all monomials $z \in M$ of weight ν (which is the weight of y). We call a monomial $z \in B^-_{\underline{i}}$ of weight ν *big* if z is bigger in the lexicographical order than all elements of M_ν. Because of the triangular form of the canonical basis it follows that $z \cdot v_\lambda = 0$ for all big $z \in B^-_{\underline{i}}$. So if y is big then set $G'(y) = y$. Otherwise, we assume that we have computed all $G'(y')$ for $y' >_{\text{lex}} y$. Write $\overline{y} - y$ as a linear combination of $z \in B^-_{\underline{i}}$ such that $z >_{\text{lex}} y$. From this linear combination we erase all big monomials, and follow the algorithm for computing $G_{\underline{i}}(y)$.

8 Remark. Here we use relations of the form $x + \sum_{x_i >_{\text{lex}} x} \zeta_i x_i \in I(\lambda)$. In the Gröbner basis algorithm the situation is reversed: there we use relations of the form $x + \sum_{x_i <_{\text{lex}} x} \zeta_i x_i \in I(\lambda)$.

9 Example. Here we use the same notation as in Example 3. Again we calculate a basis of $V(\lambda)$. The crystal graph Γ_λ is shown in Figure 2. Instead of using the paths π as points of the graph we have used the monomials x_π (cf. Lemma 7).

So the basis of $V(\lambda)$ that we get consists of elements $x \cdot v_\lambda$, where x runs through the points of the crystal graph. For instance, $v = F_{\alpha+\beta} v_\lambda$ is a basis element. We calculate $F_{\alpha+\beta} \cdot v$. We have that this is equal to $(q+q^{-1})F^{(2)}_{\alpha+\beta} v_\lambda$, which is not a basis element. So we calculate some elements $G'(x)$. All monomials of weight $2\alpha + 2\beta$ are $x_1 = F^{(2)}_\alpha F^{(2)}_\beta$, $x_2 = F_\alpha F_{\alpha+\beta} F_\beta$, $x_3 = F^{(2)}_{\alpha+\beta}$ (in decreasing lexicographical order). Here x_1 is big, which means that $G'(x_2) = x_2$. Furthermore,

$$\overline{x}_3 - x_3 = (q - q^{-1})x_2 + (q^4 - q^2 - 1 + q^{-2})x_1.$$

From this expression we discard x_1 as it is big. So we get $\overline{x}_3 - x_3 = (q - q^{-1})G'(x_2)$. Set $\delta = q$, then $\delta - \overline{\delta} = q - q^{-1}$, hence $G'(x_3) = x_3 + qx_2$. Since x_3 is not a point of the crystal graph we have $G'(x_3)v_\lambda = 0$. Therefore, $F_{\alpha+\beta} v = -(q^2 + 1)F_\alpha F_{\alpha+\beta} F_\beta v_\lambda$. (Note that we used the same relation in Example 3.)

10 Example. Again we use the notation of Example 3. As in this case the canonical basis is explicitly known, we can also construct all modules explictly. For $\nu = r\alpha + s\beta$ we let $\mathbf{B}_\nu$ be the set of all $b \in \mathbf{B}$ of weight ν. Then

$$\mathbf{B}_\nu = \begin{cases} \{F^{(r-i)}_\alpha F^{(s)}_\beta F^{(i)}_\alpha \mid 0 \le i \le r\} & \text{if } r \le s, \\ \{F^{(s-i)}_\beta F^{(r)}_\alpha F^{(i)}_\beta \mid 0 \le i \le s\} & \text{if } r > s, \end{cases}$$

(see [9], §11.17). Let $\lambda = n_1\lambda_1 + n_2\lambda_2$ be a dominant weight; we describe the action of U_q on $V(\lambda)$.

In this case it is straightforward to describe the action of the Kashiwara operators. We have

$$\widetilde{F}_\alpha(F^{(a)}_\alpha F^{(b)}_{\alpha+\beta} F^{(c)}_\beta) = F^{(a+1)}_\alpha F^{(b)}_{\alpha+\beta} F^{(c)}_\beta,$$

$$\widetilde{F}_\beta(F^{(a)}_\alpha F^{(b)}_{\alpha+\beta} F^{(c)}_\beta) = \begin{cases} F^{(a)}_\alpha F^{(b)}_{\alpha+\beta} F^{(c+1)}_\beta & \text{if } a \le c, \\ F^{(a-1)}_\alpha F^{(b+1)}_{\alpha+\beta} F^{(c)}_\beta & \text{if } a > c, \end{cases}$$

(this can easily be established using [17], 42.1.3). As before set $M = \{x_\pi \mid \pi \in \Pi_\lambda\}$. According to [16] Proposition 1.5, Corollary 2, we have that

$$M = \{\widetilde{F}^a_\alpha \widetilde{F}^b_\beta \widetilde{F}^c_\alpha(1) \mid (a, b, c) \in S^\lambda\},$$

where

$$S^\lambda = \{(a, b, c) \in \mathbb{Z}^3 \mid 0 \le c \le n_1,\ c \le b \le n_2 + c,\ 0 \le a \le n_1 + b - 2c\}.$$

11 Lemma. *For $b \ge c$ we have* $\widetilde{F}^a_\alpha \widetilde{F}^b_\beta \widetilde{F}^c_\alpha(1) = F^{(a)}_\alpha F^{(c)}_{\alpha+\beta} F^{(b-c)}_\beta$.

Proof. First of all, for $0 \leq k \leq c$ we have $\widetilde{F}_\beta^k(F_\alpha^{(c)}) = F_\alpha^{(c-k)} F_{\alpha+\beta}^{(k)}$. This follows by induction on k, along with the description of the operator $\widetilde{F}_\beta$. From this follows the case where $b = c$ of the statement $\widetilde{F}_\beta^b(F_\alpha^{(c)}) = F_{\alpha+\beta}^{(c)} F_\beta^{(b-c)}$. Now this is proved for $b \geq c$ by induction on b. It implies the statement of the lemma. $\boxed{QED}$

Lemma 11 implies that

$$M = \{F_\alpha^{(a)} F_{\alpha+\beta}^{(c)} F_\beta^{(b-c)} \mid (a,b,c) \in S^\lambda\}.$$

In the sequel we will write v_η for the vector $F_\alpha^{(a)} F_{\alpha+\beta}^{(c)} F_\beta^{(b-c)} \cdot v_\lambda$, where $\eta = (a,b,c) \in S^\lambda$. We let B_λ be the set of all v_η. By Lemma 7, B_λ is a basis of $V(\lambda)$.

12 Proposition. *Let $\eta = (a,b,c) \in S^\lambda$, then*

$$\begin{aligned} F_\alpha \cdot v_\eta =& [a+1] v_{(a+1,b,c)} \ \textit{if } a < n_1 + b - 2c \\ =& -[a+1] \sum_{k=1}^{M_1} q^{k(b-c+k)} \begin{bmatrix} a+1+k \\ k \end{bmatrix} v_{(a+1+k,b,c-k)} \\ & \textit{if } a = n_1 + b - 2c, \textit{ and } a+1+c \leq b \\ =& -[a+1] \sum_{k=1}^{M_1} q^{k(a+1+k)} \begin{bmatrix} b-c+k \\ b-c \end{bmatrix} v_{(a+1+k,b,c-k)} \\ & \textit{if } a = n_1 + b - 2c, \textit{ and } a+1+c > b, \end{aligned}$$

where $M_1 = \min(c, n_2 + c - b)$. Furthermore, set $v_0 = v_{(a,b+1,c)}$ if $b < n_2 + c$, and $v_0 = 0$ if $b = n_2 + c$. Then

$$\begin{aligned} F_\beta \cdot v_\eta &= q^{-c}[b-c+1] v_0 \ \textit{if } a = 0 \\ &= q^{a-c}[b-c+1] v_0 + [c+1] v_{(a-1,b+1,c+1)} \ \textit{if } a \geq 1 \textit{ and } c < n_1 \\ &= q^{a-c}[b-c+1] v_0 - [c+1] \sum_{k=0}^{M_2} q^{(k+1)(b+1-c+k)} \begin{bmatrix} a+k \\ k+1 \end{bmatrix} v_{(a+k,b+1,c-k)} \\ &\quad \textit{if } a \geq 1,\ c = n_1 \textit{ and } a + c \leq b+1 \\ &= q^{a-c}[b-c+1] v_0 - [c+1] \sum_{k=0}^{M_2} q^{(k+1)(a+k)} \begin{bmatrix} b-c+1+k \\ b-c \end{bmatrix} v_{(a+k,b+1,c-k)} \\ &\quad \textit{if } a \geq 1,\ c = n_1 \textit{ and } a + c > b+1, \end{aligned}$$

where $M_2 = \min(c, n_2 + c - b - 1)$.

Proof. This result is obtained by writing elements of the canonical basis as linear combinations of elements of the PBW-basis. We prove the second case (action of F_β). Firstly,

$$F_\beta v_\eta = q^{a-c}[b-c+1]F_\alpha^{(a)}F_{\alpha+\beta}^{(c)}F_\beta^{(b-c+1)}v_\lambda + [c+1]F_\alpha^{(a-1)}F_{\alpha+\beta}^{(c+1)}F_\beta^{(b-c)}v_\lambda, \quad (2)$$

where the second term is not present of $a=0$. For the first term we note that it is zero if $b=n_2+c$, as $F_\beta^{(n_2+1)}v_\lambda=0$. And if $b<n_2+c$, then it is a scalar times $v_{(a,b+1,c)}$. Suppose that $a\geq 1$. If $c<n_1$, then the second term is equal to $[c+1]$ times $v_{(a-1,c+1,b-c)}$. If $c=n_1$, then we have to distinguish two cases. The first occurs when $a+c\leq b+1$. Then the following is an element of the canonical basis:

$$F_\alpha^{(a-1)}F_\beta^{(b+1)}F_\alpha^{(c+1)} = \sum_{j=0}^{c+1} q^{(c+1+j)(b+1-j)} \begin{bmatrix} a+c-j \\ c+1-j \end{bmatrix} F_\alpha^{(a+c-j)}F_{\alpha+\beta}^{(j)}F_\beta^{(b+1-j)}$$

$$= F_\alpha^{(a-1)}F_{\alpha+\beta}^{(c+1)}F_\beta^{(b-c)} + \sum_{k=0}^{c} q^{(k+1)(b+1-c+k)} \begin{bmatrix} a+k \\ k+1 \end{bmatrix} F_\alpha^{(a+k)}F_{\alpha+\beta}^{(c-k)}F_\beta^{(b+1-c+k)}.$$

(Here we have used [9], 11.17(1).) Now this element acting on v_λ gives zero by Lemma 7 as $(a-1,b+1,c+1)\notin S^\lambda$. So using this we can rewrite the second term in (2) to a linear combination of vectors $v_{(a+k,b+1,c-k)}$. Such a vector lies in B_λ if and only if $k\leq n_2+c-b-1$. Now suppose that $k>n_2+c-b-1$, then we show that $v_{(a+k,b+1,c-k)}=0$. For that we use induction from above, with respect to the lexicographical order. By (1) we have that

$$G_{\underline{i}}(F_\alpha^{(a+k)}F_{\alpha+\beta}^{(c-k)}F_\beta^{(b+1-c+k)}) =$$
$$F_\alpha^{(a+k)}F_{\alpha+\beta}^{(c-k)}F_\beta^{(b+1-c+k)} + \sum_{s\geq 1} F_\alpha^{(a+k+s)}F_{\alpha+\beta}^{(c-k-s)}F_\beta^{(b+1-c+k+s)}.$$

Again, this element acting on v_λ gives zero. The monomials in the summation are all lexicographically bigger than $F_\alpha^{(a+k)}F_{\alpha+\beta}^{(c-k)}F_\beta^{(b+1-c+k)}$. So by induction they all map v_λ to zero. Consequently, $v_{(a+k,b+1,c-k)}=0$. Finally, if $a+c>b+1$ then we use the following element of the canonical basis:

$$F_\beta^{(c+1)}F_\alpha^{(a+c)}F_\beta^{(b-c)} = \sum_{j=0}^{c+1} q^{(c+1-j)(a+c-j)} \begin{bmatrix} b+1-j \\ b-c \end{bmatrix} F_\alpha^{(a+c-j)}F_{\alpha+\beta}^{(j)}F_\beta^{(b+1-j)} =$$

$$F_\alpha^{(a-1)}F_{\alpha+\beta}^{(c+1)}F_\beta^{(b-c)} + \sum_{k=0}^{c} q^{(k+1)(a+k)} \begin{bmatrix} b-c+1+k \\ b-c \end{bmatrix} F_\alpha^{(a+k)}F_{\alpha+\beta}^{(c-k)}F_\beta^{(b+1-c+k)}.$$

Again with the same arguments we arrive at the statement of the proposition. $\boxed{QED}$

13 Proposition. *Let* $\eta = (a,b,c) \in S^\lambda$. *Set* $v_1 = 0$ *if* $a = 0$, *and* $v_1 = [1-a-2c+b+n_1]v_{(a-1,b,c)}$ *if* $a \geq 1$, *and* $v_2 = 0$ *if* $c = 0$ *or* $b = n_2 + c$, *and* $v_2 = q^{n_1+2+b-2c}[b-c+1]v_{(a,b,c-1)}$ *otherwise. Then* $E_\alpha \cdot v_\eta = v_1 - v_2$.

PROOF. The relation

$$E_\alpha F^{(m)}_{\alpha+\beta} = F^{(m)}_{\alpha+\beta}E_\alpha - q^{-m+2}F^{(m-1)}_{\alpha+\beta}F_\beta K_\alpha$$

is easily proved by induction. We use it, along with

$$E_\alpha F^{(a)}_\alpha = F^{(a)}_\alpha E_\alpha + F^{(a-1)}_\alpha \frac{q^{1-a}K_\alpha - q^{a-1}K_\alpha^{-1}}{q-q^{-1}}$$

to show that

$$\begin{aligned} E_\alpha F^{(a)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c)}_\beta &= F^{(a)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c)}_\beta E_\alpha \\ &\quad + F^{(a-1)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c)}_\beta \frac{q^{1-a-2c+b}K_\alpha - q^{a-1+2c-b}K_\alpha^{-1}}{q-q^{-1}} \\ &\quad - q^{b-2c+2}[b-c+1]F^{(a)}_\alpha F^{(c-1)}_{\alpha+\beta}F^{(b-c+1)}_\beta K_\alpha. \end{aligned}$$

This implies the statement of the proposition. Note that the second term is not present if $a = 0$. Furthermore, the third term is not present if $c = 0$, and it maps v_λ to 0 if $b = n_2 + c$. $\boxed{QED}$

14 Proposition. *Let* $\eta = (a,b,c) \in S^\lambda$. *Set* $v_1 = 0$ *if* $b = c$, *and* $v_1 = [n_2+1-b+c]v_{(a,b-1,c)}$ *if* $b > c$, *and* $v_2 = 0$ *if* $c = 0$, *and* $v_2 = q^{2b-2c-n_2}[a+1]v_{(a+1,b-1,c-1)}$ *if* $c \geq 1$. *Then*

$$\begin{aligned} E_\beta \cdot v_\eta &= v_1 + v_2 \ \text{ if } a < n_1 + b - 2c, \text{ or } b = c, \\ &= -[n_2+1-b+c]\sum_{k=1}^{M} q^{k(a+k)} \begin{bmatrix} b-c+k-1 \\ b-c-1 \end{bmatrix} v_{(a+k,b-1,c-k)} + v_2 \\ &\quad \text{if } a = n_1 + b - 2c, \text{ and } b > c, \end{aligned}$$

where $M = \min(c, n_2 + c - b + 1)$.

PROOF. The proof is analogous to the proof of Proposition 13. This time we use the relation

$$E_\beta F^{(m)}_{\alpha+\beta} = F^{(m)}_{\alpha+\beta}E_\beta + F_\alpha F^{(m-1)}_{\alpha+\beta}K_\beta^{-1}.$$

We get

$$\begin{aligned} E_\beta F^{(a)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c)}_\beta &= F^{(a)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c)}_\beta E_\beta + \\ &\quad F^{(a)}_\alpha F^{(c)}_{\alpha+\beta}F^{(b-c-1)}_\beta \frac{q^{1-b+c}K_\beta - q^{-1+b-c}K_\beta^{-1}}{q-q^{-1}} + \\ &\quad q^{2b-2c}[a+1]F^{(a+1)}_\alpha F^{(c-1)}_{\alpha+\beta}F^{(b-c)}_\beta K_\beta^{-1}. \end{aligned}$$

If $a = n_1 + b - 2c$ then v_1 is not an element of B_λ. In that case we use $F_\beta^{(c)} F_\alpha^{(a+c)} F_\beta^{(b-1-c)} \in \mathbf{B}$ to rewrite it (note that $a + c > b - 1$). $\boxed{QED}$

4 Using tensor products

Let λ be a dominant weight, and suppose that $\lambda = \mu + \nu$, where μ, ν are both dominant weights. Set $W = V(\mu) \otimes V(\nu)$, which is a U_q-module via the comultiplication of U_q ([9], Lemma 4.8). Then the U_q-submodule of W generated by $v_\mu \otimes v_\nu$ is isomorphic to $V(\lambda)$. So, if we already have constructed $V(\mu)$, $V(\nu)$, then we can construct $V(\lambda)$ by computing the closure of $v_\mu \otimes v_\nu$ under the U_q-action. By itself this algorithm is rather inefficient for two reasons. First of all, many membership tests have to be performed in order to compute the closure. Secondly, acting on a vector $v \otimes w$, where v, w are basis elements of $V(\mu)$, $V(\nu)$ respectively, generally gives a linear combination of vectors $v' \otimes w'$. So in the end we will have rather complicated linear combinations of vectors $v \otimes w$. We cannot get around the second problem. However, the following description of a basis of $V(\lambda)$ greatly helps with the first. Here we follow [12].

Let $\pi \in \Pi_\lambda$. Then the first direction of π is $w(\lambda)$ for some $w \in W/W_\lambda$, where W_λ is the stabilizer of λ ([14], 5.2). Set $\phi(\pi) = w$. Let $s_{i_1} \cdots s_{i_r}$ be the reduced expression for $\phi(\pi)$, which is lexicographically the smallest. (Here $s_{i_1} \cdots s_{i_r} <_{\mathrm{lex}} s_{j_1} \cdots s_{j_r}$ if there is a $k > 0$ such that $i_1 = j_1, \ldots, i_{k-1} = j_{k-1}$ and $i_k < j_k$.) Then we define integers $n_1, \ldots, n_r$, and paths $\pi_0, \pi_1, \ldots, \pi_r$ in the following way. First, $\pi_0 = \pi$. We let n_k be maximal such that $e_{\alpha_{i_k}}^{n_k} \pi_{k-1} \neq 0$, and we set $\pi_k = e_{\alpha_{i_k}}^{n_k} \pi_{k-1}$. Set $F_\pi = F_{\alpha_{i_1}}^{(n_1)} \cdots F_{\alpha_{i_r}}^{(n_r)}$. Then the set $\{F_\pi \cdot v_\lambda \mid \pi \in \Pi_\lambda\}$ is a basis of $V(\lambda)$ ([12]).

Now let $\pi \in \Pi_\lambda$, and let $\phi(\pi) = s_{i_1} \cdots s_{i_r}$ be the reduced expression which is the smallest in the lexicographical order. Let $F_\pi = F_{\alpha_{i_1}}^{(n_1)} \cdots F_{\alpha_{i_r}}^{(n_r)}$. Write $\alpha = \alpha_{i_1}$. If $n_1 > 1$, then $\phi(e_\alpha \pi) = \phi(\pi)$, and hence $F_{e_\alpha \pi} = F_{\alpha_{i_1}}^{(n_1-1)} \cdots F_{\alpha_{i_r}}^{(n_r)}$. On the other hand, if $n_1 = 1$, then $\phi(e_\alpha \pi) = s_{i_2} \cdots s_{i_r}$ which is the smallest (in the lexicographical order) reduced expression for $\phi(e_\alpha \pi)$. Hence $F_{e_\alpha \pi} = F_{\alpha_{i_2}}^{(n_2)} \cdots F_{\alpha_{i_r}}^{(n_r)}$. The conclusion is that

$$F_\pi \cdot v_\lambda = \frac{1}{[n_1]_\alpha} F_\alpha \cdot (F_{e_\alpha \pi} \cdot v_\lambda).$$

So in order to compute $F_\pi \cdot v_\lambda$, we only have to act with F_α on a vector that we already computed. Hence we have a direct algorithm to compute a basis of the submodule generated by $v_\mu \otimes v_\nu$.

15 Remark. With this algorithm we can construct any highest-weight module, provided that we know how to construct the fundamental modules $V(\lambda_i)$

(where λ_i is a fundamental weight). We can construct those modules using a different algorithm, such as the Gröbner basis algorithm of Section 2. Alternatively, in some cases we can use known constructions. For example, if the root system is of type A_n, then all fundamental weights are minuscule, and the corresponding modules can be constructed as in [9], §5A.1. Also [13] contains a description of the fundamental modules when the type of the root system is C_n.

16 Example. Again we use the same notation as in Example 3. We construct $V(\lambda)$ as a submodule of $V(\lambda_1) \otimes V(\lambda_2)$. From the crystal graph (Figure 2) we see that the set Π_λ consists of the paths π_λ, $f_\alpha\pi_\lambda$, $f_\beta\pi_\lambda$, $f_\beta f_\alpha\pi_\lambda$, $f_\alpha f_\beta\pi_\lambda$, $f_\beta^2 f_\alpha\pi_\lambda$, $f_\alpha^2 f_\beta\pi_\lambda$, $f_\alpha f_\beta^2 f_\alpha\pi_\lambda$. They correspond to the reduced expressions 1, s_α, s_β, $s_\beta s_\alpha$, $s_\alpha s_\beta$, $s_\beta s_\alpha$, $s_\alpha s_\beta$, $s_\alpha s_\beta s_\alpha$. Set $w = v_{\lambda_1} \otimes v_{\lambda_2}$, and let W be the U_q-submodule of $V(\lambda_1) \otimes V(\lambda_2)$ generated by w. Then W is isomorphic to $V(\lambda)$. Furthermore, by the above, a basis of W is given by w, $F_\alpha w$, $F_\beta w$, $F_\beta F_\alpha w$, $F_\alpha F_\beta w$, $F_\beta^{(2)} F_\alpha w$, $F_\alpha^{(2)} F_\beta w$, $F_\alpha F_\beta^{(2)} F_\alpha w$. Now, since λ_1, λ_2 are minuscule, the modules $V(\lambda_1)$, and $V(\lambda_2)$ are easy to construct. So we can express the basis elements above as linear combinations of $v_1 \otimes v_2$, where v_1, v_2 are basis elements of $V(\lambda_1)$, $V(\lambda_2)$ respectively. This then allows us to compute the action of any element of U_q with respect to this basis.

5 Using the dual space

The antipode of U_q is an anti-automorphism $S : U_q \to U_q$ given by $S(E_\alpha) = -K_\alpha^{-1} E_\alpha$, $S(F_\alpha) = -F_\alpha K_\alpha$, $S(K_\alpha) = K_\alpha^{-1}$ (see [9], 4.8). We have that U_q acts on the dual space $M(\lambda)^*$ via the antipode S, i.e., $u \cdot f(v) = f(S(u) \cdot v)$.

By λ^* we denote the dominant weight such that $w_0(\lambda) = -\lambda^*$ (where w_0 is the longest element in the Weyl group).

In the sequel we denote by 1 and v_λ the image of $1 \in U_q$ in $M(\lambda)$ and $V(\lambda)$ respectively.

Now let $W = \{f \in M(\lambda)^* \mid f(I(\lambda)) = 0\}$. Then it is clear that W is a submodule of $M(\lambda)^*$ with $\dim W = \dim V(\lambda)$. Set $\mu = -\lambda^*$ and $\nu = \lambda - \mu$. Let $f_\mu \in M(\lambda)^*$ be defined as follows. First we note that the weight space $V(\lambda)_\mu$ is 1-dimensional. We choose an $a \in U^-$ of weight ν such that $a \notin I(\lambda)$. Then $a \cdot v_\lambda$ spans $V(\lambda)_\mu$. We set $f_\mu(a \cdot 1) = 1$, and $f_\mu(I(\lambda)) = 0$. Furthermore, $f_\mu(x \cdot 1) = 0$ for all homogeneous $x \in U^-$, not of weight ν. Now, since $S(K_\alpha) = K_\alpha^{-1}$ we see that f_μ is a weight vector of weight $-\mu = \lambda^*$. Also $E_\alpha \cdot f_\mu(x \cdot 1) = -f_\mu(K_\alpha^{-1} E_\alpha x \cdot 1)$. Now $K_\alpha^{-1} E_\alpha x \cdot 1$ is a weight vector of weight μ if and only if $x \in U^-$ is of weight $\nu + \alpha$. But that means that $x \in I(\lambda)$, and hence also $K_\alpha^{-1} E_\alpha x \in I(\lambda)$. It follows that $E_\alpha \cdot f_\mu = 0$ for $\alpha \in \Delta$. Therefore f_μ is a highest-weight vector. So $U_q \cdot f_\mu$ is a U_q-submodule of W isomorphic to $V(\lambda^*)$. But $V(\lambda^*) \cong V(\lambda)^*$ ([9], Proposition

5.16). By comparing dimensions we see that $W \cong V(\lambda^*)$. In the sequel we will outline how to construct a basis of W, along with the action of a $u \in U_q$ on W.

Let η be a dominant weight, and let $v_0 \in V(\eta)$ be a non-zero vector of weight $-\eta^*$. Then v_0 is called a lowest-weight vector of $V(\eta)$; it satisfies $F_\alpha \cdot v_0 = 0$ for $\alpha \in \Delta$.

Let $f_0 \in M(\lambda)^*$ be defined by $f_0(1) = 1$, $f_0(u \cdot 1) = 0$ if $u \in U^-$ is a homogeneous element, $\neq 1$. Then $f_0 \in W$, and f_0 is a weight vector of weight $-\lambda$. Furthermore, $F_\alpha \cdot f_0 = 0$ so that f_0 is a lowest weight vector. This implies that $W = U^+ \cdot f_0$.

Now let $\omega : U_q \to U_q$ be the automorphism defined by $\omega(F_\alpha) = E_\alpha$, $\omega(E_\alpha) = F_\alpha$, $\omega(K_\alpha) = K_\alpha^{-1}$ (cf. [9], 4.6).

17 Lemma. *Suppose that $\{m_1 \cdot v_\lambda, \dots, m_t \cdot v_\lambda\}$ is a basis of $V(\lambda)$, where $m_i \in U^-$. Let v_0 be a lowest-weight vector of $V(\lambda^*)$. Then $\{\omega(m_1) \cdot v_0, \dots, \omega(m_t) \cdot v_0\}$ is a basis of $V(\lambda^*)$.*

Proof. Set $J^+(\lambda) = \omega(J(\lambda))$; it is the left-ideal of U_q generated by F_α, $K_\alpha^{\pm 1} - q^{\mp(\alpha,\lambda)}$. Set $M^+(\lambda) = U_q/J^+(\lambda)$, which is a U_q-module. Then ω induces a bijective linear map $\omega : M(\lambda) \to M^+(\lambda)$ such that $u \cdot \omega(a) = \omega(\omega(u) \cdot a)$ for $u \in U_q$ and $a \in M(\lambda)$. This implies that $I^+(\lambda) = \omega(I(\lambda))$ is a U_q-submodule of $M^+(\lambda)$. Also set $V^+(\lambda) = M^+(\lambda)/I^+(\lambda)$; then $\dim V^+(\lambda) = \dim V(\lambda)$. Also ω induces a bijective linear map $\omega : V(\lambda) \to V^+(\lambda)$, such that $u \cdot \omega(v) = \omega(\omega(u) \cdot v)$ for $u \in U_q$ and $v \in V(\lambda)$. Now let $u_0 \in V(\lambda)$ be a vector of weight $-\lambda^*$, i.e., a lowest-weight vector. Then $K_\alpha \cdot \omega(u_0) = \omega(K_\alpha^{-1} \cdot u_0)$, from which we see that $\omega(u_0)$ is of weight λ^*. Also $E_\alpha \cdot \omega(u_0) = \omega(F_\alpha \cdot u_0) = 0$ and hence $\omega(u_0)$ is a highest-weight vector of $V^+(\lambda)$ of weight λ^*. It follows that $V^+(\lambda) \cong V(\lambda^*)$. In the same way we see that $\omega(v_\lambda)$ is a lowest-weight vector of $V^+(\lambda)$. Let v_0 also denote the image of $v_0 \in V(\lambda^*)$ under the isomorphism $V^+(\lambda) \cong V(\lambda^*)$. Then v_0 is a scalar multiple of $\omega(v_\lambda)$. The result follows. $\boxed{QED}$

Let F_π be as in Section 4, and set $E_\pi = \omega(F_\pi)$. Then by Lemma 17 the elements $E_\pi \cdot f_0$, where π runs through Π_λ, form a basis B_W of W.

So we have a basis of W. But in order to compute the matrix of the action of a $u \in U_q$ on W we need to be able to express $u \cdot f$ (for $f \in B_W$) as a linear combination elements of B_W. We do that as follows. Let $x_1, \dots, x_t$ be the elements of $B_{\underline{i}}^-$ obtained by the procedure of Lemma 7. This means that $\{x_i \cdot v_\lambda\}$ is a basis of $V(\lambda)$. To an element $f \in W$ we associate the row vector $c_f = (f(x_1), \dots, f(x_t))$. Then the map $f \mapsto c_f$ is a bijective linear map from W to $\mathbb{Q}(q)^t$. So computing the vector $(u \cdot f(x_1), \dots, u \cdot f(x_t))$ allows us to express $u \cdot f$ as a linear combination of elements of B_W.

18 Remark. In the algorithm we need to compute values of $f \in B_W$. Suppose that $f = E_\pi \cdot f_0$, where $E_\pi = E_{\alpha_{i_1}}^{(n_1)} \cdots E_{\alpha_{i_r}}^{(n_r)}$. Then as in Section 4 we have

$$E_\pi \cdot f_0 = \frac{1}{[n_1]_\alpha} E_\alpha \cdot (E_{e_\alpha \pi} \cdot f_0),$$

where $\alpha = \alpha_{i_1}$. So we see that when computing values of $E_\pi \cdot f_0$ we can use the values of $E_{e_\alpha \pi} \cdot f_0$, which we also need.

19 Example. We use the notation from Example 3. We let W denote the U_q-submodule of $M(\lambda)^*$ generated by f_0. Then using Example 16 we see that W has a basis consisting of the elements f_0, $E_\alpha f_0$, $E_\beta f_0$, $E_\beta E_\alpha f_0$, $E_\alpha E_\beta f_0$, $E_\beta^{(2)} E_\alpha f_0$, $E_\alpha^{(2)} E_\beta f_0$, $E_\alpha E_\beta^{(2)} E_\alpha f_0$. Set $f_\lambda = E_\alpha E_\beta^{(2)} E_\alpha f_0$; we compute $F_{\alpha+\beta} \cdot f_\lambda$ with respect to the basis above. For that we have to compute the values of elements of $M(\lambda)^*$ when evaluated on the monomials 1, F_α, F_β, $F_{\alpha+\beta}$, $F_\alpha F_\beta$, $F_{\alpha+\beta} F_\beta$, $F_\alpha^{(2)} F_\beta$, $F_\alpha F_{\alpha+\beta} F_\beta$ (cf. Example 9). For $F_{\alpha+\beta} \cdot f_\lambda$ this leads to the vector $(0, 0, 0, q^{-4} + 1, -q^{-1}, 0, 0, 0)$. Also, the functions $E_\beta E_\alpha f_0$ and $E_\alpha E_\beta f_0$ correspond to $(0, 0, 0, q^{-4}, q^{-3}, 0, 0, 0)$ and $(0, 0, 0, -q^{-1}, q^{-4}+q^{-2}, 0, 0, 0)$ respectively. Hence $F_{\alpha+\beta} \cdot f_\lambda = E_\beta E_\alpha f_0 - q E_\alpha E_\beta f_0$. The calculations necessary for obtaining the vectors above were done in GAP (cf. [7]).

6 Gelfand-Zetlin patterns

Here we work with $U_q(\mathfrak{sl}_n)$, and we use ideas from [10], [19]. Let $\lambda = r_1\lambda_1 + \cdots + r_{n-1}\lambda_{n-1}$ be a dominant weight. Then a Gelfand-Zetlin pattern is a tableau of the form

$$m = \begin{array}{ccccccccc} m_{1,n} & & m_{2,n} & & m_{3,n} & \cdots & & & m_{n,n} \\ & m_{1,n-1} & & m_{2,n-1} & \cdots & \cdots & m_{n-1,n-1} & & \\ & & & \vdots & & & & & \\ & & m_{1,2} & & m_{2,2} & & & & \\ & & & m_{1,1} & & & & & \end{array}$$

such that

(1) $m_{n,n} = 0$ and $m_{i,n} - m_{i+1,n} = r_i$,

(2) $m_{i,j+1} \geq m_{i,j} \geq m_{i+1,j+1}$.

Let m be a Gelfand-Zetlin pattern. Then we let $m^\pm[i,j]$ be the pattern which is equal to m, except on position (i,j) where there is $m_{i,j} \pm 1$. Also we set $\widehat{m}_{i,j} = m_{i,j} - i$. Furthermore, if $m^\pm[j,k]$ is not a Gelfand-Zetlin pattern then we set $a^\pm_{j,k}(m) = 0$, and otherwise

$$a^\pm_{j,k}(m) = \mp \frac{\prod_{i=1}^{k\pm 1} [\widehat{m}_{i,k\pm 1} - \widehat{m}_{j,k}]}{\prod_{i=1, i\neq j}^{k} [\widehat{m}_{i,k} - \widehat{m}_{j,k}]}$$

(where $[k]$ is the same as $[k]_\alpha$ with $q_\alpha = q$). Then we define an action of U_q on the vector space spanned by all Gelfand-Zetlin patterns by

$$\begin{aligned} F_{\alpha_k} \cdot m &= \sum_{j=1}^{k} a^-_{j,k}(m) m^-[j,k] \\ E_{\alpha_k} \cdot m &= \sum_{j=1}^{k} a^+_{j,k}(m) m^+[j,k] \\ K_{\alpha_k} \cdot m &= q^{b_k} m, \end{aligned}$$

where

$$b_k = -\sum_{i=1}^{k-1} m_{i,k-1} + 2\sum_{i=1}^{k} m_{i,k} - \sum_{i=1}^{k+1} m_{i,k+1} \text{ for } 1 \le k \le n-1.$$

20 Remark. The formulas above are taken from [19]. The formula for the action of K_α had to be changed slightly, because in that paper the authors dealt with $U_q(\mathfrak{gl}_n)$, wheras we work with $U_q(\mathfrak{sl}_n)$.

21 Example. We use the notation from Example 3. In this case the Gelfand-Zetlin patterns are

$$\begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 1 & \\ & & 2 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 1 & \\ & & 1 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 0 & \\ & & 2 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 0 & \\ & & 1 & & \end{array},$$

$$\begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 0 & \\ & & 0 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 1 & & 1 & \\ & & 1 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 1 & & 0 & \\ & & 1 & & \end{array}, \quad \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 1 & & 0 & \\ & & 0 & & \end{array}.$$

We use the formulas above to compute the action of elements of U_q. For example

$$E_\beta \cdot \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 1 & & 0 & \\ & & 1 & & \end{array} = \frac{1+q^2+q^4}{q+q^3} \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 2 & & 0 & \\ & & 1 & & \end{array} + \frac{1+q^2+q^4}{q+q^3} \begin{array}{ccccc} 2 & & 1 & & 0 \\ & 1 & & 1 & \\ & & 1 & & \end{array}.$$

7 Practical experiences

I have implemented the algorithms described in this paper in the computer algebra system GAP4.3 ([3]), using the package QuaGroup ([7]).

Table 1 contains the running time (in seconds) of the algorithms on a few sample inputs. The algorithms of Sections 2, 3, 4, 5, and 6 are denoted GB, CB, TP, DM, and GZ respectively. The calculations were done on a Pentium III, 600MHz processor, with 64M working memory for GAP.

The input to all algorithms is the same; however every algorithm gives different output. They all produce a module, but the function for computing the action of an element of U_q is different. So, in order to compare the algorithms, the canonical basis of the module is computed, as well as the matrices of the generators of U_q with respect to that basis. In this way all algorithms give exactly the same output, and the running times can be compared fairly. The algorithm for computing the canonical basis of a module that was used, is described in [8].

22 Remark. In the implementation of the algorithm using tensor products all weights are written as a sum of two weights. For most cases, there is only one possibility for this. Furthermore, I have used $(0,1,2,0) = (0,1,0,0) + (0,0,2,0)$ and $(2,1) = (2,0)+(0,1)$. The two modules corresponding to the smaller weights were computed using the Gröbner basis algorithm.

On the performance of the algorithms I remark the following:

- The Gröbner basis algorithm runs quite fast, except in two cases (F_4 with $(0,0,0,2)$ and G_2 with $(2,1)$). The exceptionally high running times in these examples are caused by an explosion of the sizes of the elements of $\mathbb{Q}(q)$ that the program has to deal with. This happens when elements of the Gröbner basis are reduced modulo each other. The resulting elements always have nice coefficients, but the coefficients of the intermediate elements become quite big. This problem occurs more acutely in the case of F_4, and G_2 because in those cases the commutation relations that are used to compute products are relatively "dense" (i.e., contain many monomials with nonzero coefficient).

- The algorithm using canonical bases is seen to be rather inefficient. Many elements of the canonical basis have to be computed, and this number increases very rapidly with the dimension of the module. For example, in the case of A_4, with $(1,1,1,0)$ the algorithm had to compute 1211 elements of the canonical basis, and for the weight $(0,1,2,0)$ this was already 2987. The bottleneck here is the computation of $\overline{x}$ for $x \in B^-_{\underline{i}}$.

- The algorithm using tensor products behaves quite well. Generally it is not as fast as the Gröbner basis algorithm, but it also manages to beat the latter on two occasions. This is seen most spectacularly in the case of G_2 with $(2,1)$, where the tensor product algorithm does not suffer from coefficient blow-up.

- The algorithm that uses the dual module $M(\lambda)^*$ struggles a lot. This is caused by the fact that many images of functions have to be calculated, for which it is necessary to calcuate many products of elements of U_q. This is also illustrated by Example 19.

- Finally, the method using Gelfand-Zetlin patterns performs quite well for the case of A_n. In this case, producing a basis for the module is very fast. However, the algorithm loses time when computing the canonical basis of the module. This is caused by the fact that the coefficients given by the formulas in Section 6 are quite "ugly" rational functions.

We conclude that for practical purposes the algorithms GB, TP, and GZ appear to be the most efficient. However, GZ only works when the root system is of type A_n. Furthermore, TP can beat GB in examples where the latter suffers from coefficient blow-up.

type	λ	$\dim V(\lambda)$	GB	CB	TP	DM	GZ
A_4	$(0,1,1,0)$	75	24	54	48	234	37
A_4	$(1,1,1,0)$	280	192	742	372	5297	312
A_4	$(0,1,2,0)$	315	247	2283	446	6744	367
C_4	$(0,0,1,0)$	48	20	68	$\times$	206	$\times$
C_4	$(1,1,0,0)$	160	134	∞	245	2033	$\times$
C_4	$(0,2,0,0)$	308	424	∞	657	10684	$\times$
F_4	$(0,0,0,1)$	26	14	48	$\times$	191	$\times$
F_4	$(1,0,0,0)$	52	40	∞	$\times$	818	$\times$
F_4	$(0,0,0,2)$	324	3428	∞	1157	∞	$\times$
G_2	$(0,1)$	14	3	4	$\times$	22	$\times$
G_2	$(1,1)$	64	81	99	38	467	$\times$
G_2	$(2,1)$	189	11566	∞	261	∞	$\times$

Table 1. Running times of the algorithms of Sections 2-6, in seconds. A weight is represented by giving its coefficients when written as a linear combination of fundamental weights; for this the same ordering of simple roots was used as in [1]. A $\times$ means that the algorithm is not applicable. A ∞ indicates that the running time was more than 4 hours.

References

[1] N. Bourbaki: *Groupes et Algèbres de Lie, Chapitres 4, 5 et 6.* Hermann, Paris, 1968.

[2] C. De Concini and C. Procesi: *Quantum groups.* In D-modules, representation theory, and quantum groups (Venice, 1992), p. 31–140. Springer, Berlin, 1993.

[3] The GAP Group. *GAP – Groups, Algorithms, and Programming*, Version 4.3, 2002. (`http://www.gap-system.org`).

[4] W. A. de Graaf: Lie Algebras: Theory and Algorithms, volume 56 of North-Holland Mathematical Library. Elsevier Science, 2000.

[5] W. A. de Graaf: *Computing with quantized enveloping algebras: PBW-type bases, highest-weight modules, R-matrices.* J. Symbolic Comput., **32**(5): p. 475–490, 2001.

[6] W. A. de Graaf: *Constructing representations of split semisimple Lie algebras.* J. Pure Appl. Algebra, **164**(1-2): p. 87–107, 2001. Effective methods in algebraic geometry (Bath, 2000).

[7] W. A. de Graaf: QuaGroup. A GAP share package, 2001. (`http://www.math.uu.nl/people/graaf/quagroup.html`).

[8] W. A. de Graaf: *Constructing canonical bases of quantized enveloping algebras.* Experimental Mathematics, page to appear, 2002.

[9] J. C. Jantzen: Lectures on Quantum Groups, volume 6 of Graduate Studies in Mathematics. American Mathematical Society, 1996.

[10] M. Jimbo: *Quantum R matrix related to the generalized Toda system: an algebraic approach.* In Field theory, quantum gravity and strings (Meudon/Paris, 1984/1985), p. 335–361. Springer, Berlin, 1986.

[11] M. Kashiwara: *Similarity of crystal bases.* In Lie algebras and their representations (Seoul, 1995), p. 177–186. Amer. Math. Soc., Providence, RI, 1996.

[12] V. Lakshmibai: *Bases for quantum Demazure modules.* In Representations of groups (Banff, AB, 1994), p. 199–216. Amer. Math. Soc., Providence, RI, 1995.

[13] C. Lecouvey: *An algorithm for computing the global basis of an irreducible $U_q(\mathfrak{sp}_{2n})$-module.* Adv. in Appl. Math., **29**(1): p. 46–78, 2002.

[14] P. Littelmann: *A Littlewood-Richardson rule for symmetrizable Kac-Moody algebras.* Invent. Math., **116**(1-3): p. 329–346, 1994.

[15] P. Littelmann: *Paths and root operators in representation theory.* Ann. of Math. (2), **142**(3): p. 499–525, 1995.

[16] P. Littelmann: *Cones, crystals, and patterns.* Transform. Groups, **3**(2): p. 145–179, 1998.

[17] G. Lusztig: Introduction to quantum groups. Birkhäuser Boston Inc., Boston, MA, 1993.

[18] G. Lusztig: *Braid group action and canonical bases.* Adv. Math., **122**(2): p. 237–261, 1996.

[19] V. Mazorchuk and L. Turowska: *On Gelfand-Zetlin modules over $U_q(\mathrm{gl}_n)$.* Czechoslovak J. Phys., **50**(1): p. 139–144, 2000. Quantum groups and integrable systems (Prague, 1999).