1. Introduction

The importance of the Lie theory for applied mathematics and the physical sciences has grown substantially in recent years. In part, the renewed interest in the Lie theory can be attributed directly to the development of improved computational methods. Modern methods permit many important calculations involving the irreducible representations of semisimple Lie algebras to be carried out in a systematic and uniform way on an electronic digital computing machine using integer mode arithmetic. We present here an historical survey of the use of computers in Lie algebra theory, with particular reference to computing the coupling and recoupling coefficients for the irreducible representations of simple Lie algebras of arbitrary type using Chevalley bases.

In addition to the intrinsic mathematical interest in the coupling and recoupling coefficients, an important motivation for computing these numbers comes from their extensive applications in atomic, nuclear and elementary particle physics. There already exists an extensive literature on the problem of computing these coefficients, and progress continues to be made. Yet, despite this impressive literature, a complete resolution of the problem does not yet exist in print. The coupling coefficients needed in applications can often be computed using various special tricks, but no generally applicable algorithm for doing this has been published.
We shall have achieved our aim if, in presenting the following survey of the literature on this problem, we have enabled others to perceive at least the general outline for such a universal algorithm by which one may compute the coupling coefficients for any simple Lie algebra over the complex number field.

For convenience, we have arranged the material in chronological order, based on publication dates. In many cases, of course, the actual work was done a year or two earlier. We recognize that the bibliography is not complete, and we hereby offer our apologies to those whose work has been left out.

2. Classical Work, Before 1960

E. B. Dynkin in 1947 published an exposition of Lie algebra theory starting from first principles [1]. This survey, based on the classical work of S. Lie, W. Killing, E. Cartan, H. Weyl and A. I. Mal'cev, contains important algorithms for Lie algebras still used today. A new feature in Dynkin's treatment was the introduction of schemas, now called Dynkin diagrams, which summarize information about the angles and relative lengths of a system of simple roots of a semisimple Lie algebra over the field of complex numbers.

Let us denote by \( r \) the rank of a semisimple complex Lie algebra. By definition, the rank is the dimension of any Cartan subalgebra. The simple roots, \( \alpha_1, \alpha_2, \ldots, \alpha_r \), are certain nonzero linear forms on a Cartan subalgebra. For any nonzero linear form \( \alpha \) on the Cartan subalgebra, we define another form \( \hat{\alpha} \) by

\[
\hat{\alpha} = \frac{2\alpha}{(\alpha, \alpha)}. \]
The Dynkin diagram determines an integer array \( M_{ij} \) which is equal to a numerical factor times \((\alpha_i, \alpha_j)\). The unknown numerical factor drops out when we compute the integer Cartan matrix

\[
A_{ij} = (\alpha_i, \alpha_j) = 2 M_{ij} / M_{ii}.
\]

The entire Lie algebra can be reconstructed from the Cartan matrix by classical algorithms explained in Dynkin's paper.

The irreducible finite-dimensional representations of certain semisimple Lie algebras were studied in 1950 by I. M. Gel'fand and M. L. Zetlin using explicit formulas based on chains of subalgebras [2,3]. To generalize this method to other Lie algebras, it is necessary to study the subalgebra structure of a given semisimple Lie algebra.

In 1952, such a study of subalgebras was initiated by E. B. Dynkin [4]. In the appendix to his paper on maximal subalgebras, Dynkin summarized the main facts of the theory of irreducible representations of semisimple Lie algebras over the complex number field. Each irreducible representation is characterized by its highest weight \( \lambda \). Any weight \( \mu \) is an integer linear combination of the highest weights \( \lambda_1, \lambda_2, \ldots, \lambda_k \) of certain basic representations,

\[
\mu = m_1 \lambda_1 + m_2 \lambda_2 + \ldots + m_k \lambda_k
\]

These basic weights \( \lambda_1, \lambda_2, \ldots, \lambda_k \) are geometrically determined by the condition that \((\alpha_i, \lambda_j) = \delta_{ij}\) is the Kronecker delta, equal to one if \( i = j \) and zero if \( i \neq j \). The integer coefficients

\[
m_i = (\mu, \alpha_i)
\]

are called the Dynkin indices of the weight \( \mu \). It is convenient in computational work to express all roots and weights in terms of their Dynkin indices. In particular, the Cartan matrix represents the Dynkin indices of the simple roots. The
Dynkin indices of the highest weight of any representation are non-negative integers, and conversely, to any set of non-negative integers \((n_1, n_2, \ldots, n_k)\) there corresponds, up to equivalence, a unique irreducible representation with highest weight \(n_1 \lambda_1 + n_2 \lambda_2 + \cdots + n_k \lambda_k\). All information about an irreducible representation can be expressed in terms of the highest weight. In particular, there are two formulas due to H. Weyl which can be used to compute the dimension and the value of the second order Casimir operator.

In this same appendix, Dynkin describes an algorithm for computing the entire weight system of an irreducible representation from its highest weight. The weight system is built up layer by layer, each weight on a given layer being obtained from some weight on the preceding layer by subtracting some simple root. For the total number of layers, Dynkin gives a formula, which can be written as

\[
1 + 2 \sum_{i,j} n_i (A^{-1})_{ij}.
\]

Here the \(n_i\) are the Dynkin indices of the highest weight, while \((A^{-1})_{ij}\) are the entries of the inverse of the Cartan matrix. The inverse of the Cartan matrix can also be used to compute the metric tensor

\[
g_{ij} = \langle \lambda_i, \lambda_j \rangle = \frac{1}{2} (\alpha_i, \alpha_j) (A^{-1})_{ij}.
\]

This metric tensor is needed to compute inner products of weights when we express them in terms of Dynkin indices. Note that since the Dynkin diagram only gives the relative lengths of the simple roots, the metric tensor is as yet known only up to an overall factor. We can determine this factor by using the fact that the Casimir operator has the value one in the adjoint representation.

Another paper by Dynkin, also written in 1952, deals with semisimple subalgebras [5]. In the introduction to this paper,
explicit tables are given for the metric tensor $g_{ij} = (\lambda_i, \lambda_j)$ and its inverse $g^{ij} = (\hat{\alpha}_i, \hat{\alpha}_j)$.

A new algorithm for computing the characters of the irreducible representations of semisimple Lie algebras was developed by H. Freudenthal in 1954. This procedure is more efficient than an older Weyl formula using 'girdle division.' The essence of Freudenthal's method lies in the construction of his 'table D' and the use of an inductive formula for computing the multiplicities of the dominant weights on a given layer in terms of the multiplicities of the weights on previous layer [6]. By means of this procedure, the characters of even the exceptional simple Lie algebra $E_8$ could be obtained by hand. This algorithm remains today the quickest way to compute characters.

A valuable computational tool for Lie algebras has grown out of the discovery by C. Chevalley in 1955 of a new class of finite simple groups related to Lie algebras [7]. Chevalley found a systematic construction of bases for Lie algebras with respect to which all calculations can be carried out using integer-mode arithmetic. The structure constants are integers computed as follows. First, for each nonzero root $\alpha$, a co-root $h_\alpha$ is defined. This co-root belongs to the Cartan subalgebra and satisfies

$$(h_\alpha, h) = \hat{\alpha}(h)$$

for all elements $h$ in the Cartan subalgebra. Next, Chevalley shows that one can assign a root vector $x_\alpha$ to each non-zero root $\alpha$ such that

$$[h, x_\alpha] = \alpha(h) x$$
$$[x_\alpha, x_\beta] = h_\gamma$$

and

$$[x_\alpha, x_\beta] = N_{\alpha, \beta} x_{\alpha + \beta}$$
where the integers \( N_{\alpha, \beta} \) satisfy \( N_{-\alpha, -\beta} = -N_{\alpha, \beta} \). These integers are determined up to sign by

\[
N_{\alpha, \beta} = \pm (p+1),
\]

where \( \beta = \rho_a, \ldots, \beta + qa \) is the \( \alpha \)-string through the root \( \beta \).

To help dispel some of the possibly mysterious aspects surrounding the use of Chevalley bases, we consider the most elementary example, the simple Lie algebra \( A_1 \). This is the Lie algebra studied in the quantum theory of angular momentum. It is the complexification of the real Lie algebra of the Lie group \( SU(2) \). Here one has only a single positive root \( \alpha \), and we may set \( e = x_\alpha, f = x_{-\alpha} \) and \( h = h_\alpha \). The Lie products are

\[
[e, f] = h \\
[h, e] = \pm e \\
[h, f] = -2f.
\]

The connection with the notation used in the quantum theory of angular momentum is given by the formulas

\[
h = 2j_3 \\
e = j_1 + \sqrt{-1} j_2 \\
f = j_1 - \sqrt{-1} j_2.
\]

We call attention here to the factor 2 in the equation \( h = 2j_3 \) which serves to eliminate all the half-integers abounding in angular momentum theory.

R. Bivins, N. Metropolis, M. Rotenberg and J. K. Wooten, Jr. used electronic computers to prepare extensive tables of Clebsch-Gordan and Racah coefficients for \( SU(2) \). These tables, published in 1959, are used by physicists and chemists in a wide variety of fields [8].

It was also in the year 1959 that B. Kostant published his elegant formula for the multiplicity of a weight, using a certain partition function [9]. This closed formula, like the
Weyl formula, involves summing over the Weyl group, but it
does not require division. The obvious question arises whether
this new formula is an improvement over the Freudenthal algor-
ithm.

3. Work in the 1960's

R. Steinberg in 1961 used the Kostant formula to develop
a formula for the reduction of the tensor product of two irredu-
cible modules over a semisimple Lie algebra as a direct sum of
irreducible submodules [10]. This expression for the multi-
plicities of the irreducible submodules occurring in the reduc-
tion involves the partition function and a double summation over
the Weyl group. The Steinberg formula reduces to the original
Clebsch-Gordan results when one specializes to the simple Lie
algebra $A_1$. The work of Steinberg of course concerns only the
Clebsch-Gordan series, not the Clebsch-Gordan coefficients.

In that same year, physicists, led by M. Gell-Mann, found
experimental evidence that the Lie group SU(3) is an approxi-
mate symmetry in the dynamics of the strongly interacting
elementary particles [11]. This work created widespread
interest in this particular Lie group as well as in the appli-
cation of Lie group theory in general.

In the following year, N. Jacobson published the first
book on Lie algebras in the English language [12]. Many of
the above-mentioned algorithms for Lie algebraic calculations
are described in detail in this book.

Also in 1962 an interesting monograph by I. B. Levinson,
V. V. Vanagas and A. P. Yutsis appeared in which a graphical
calculus for handling Clebsch-Gordan and Racah coefficients
for SU(2) was developed [13]. This technique allows one to
draw pictures giving insight into the algebraic formulas.
Extensive hand calculations were done in 1963 by M. Konuma, K. Shima and M. Wada, and by other physicists, for the rank 2 and rank 3 simple Lie algebras needed in applications [14]. The needs of particle physics also led J. J. de Swart to compute the Clebsch–Gordan coefficients for SU(3) by hand [15].

By 1964 the computation of the coupling and recoupling coefficients for the group SU(2) had progressed to the point that representations of dimension up to a hundred could be handled. R. M. Baer and M. G. Redlich reported that such large integers occur in these calculations that it became necessary to introduce multiple precision fixed point arithmetic subroutines [16].

While studying extensions of Chevalley's method of constructing finite simple groups, R. Ree in 1964 introduced the concept of a Chevalley basis for a Lie module [17]. With respect to such a basis, the action of the Lie algebra on the module can be described by integer matrices. Ree first proved the existence of Chevalley bases for certain elementary irreducible representations for each type of simple Lie algebra, and then used Cartan composition to extend the result to arbitrary irreducible representations.

D. A. Smith gave a new proof of the existence of Chevalley bases for Lie modules in 1965 which avoids Cartan composition and the numerous case considerations required in Ree's proof. Instead, he extends Chevalley's formulas for the Lie algebra to its universal enveloping associative algebra, and uses the fact that every irreducible module is a cyclic module over the enveloping algebra [18].

The application of Chevalley bases to the construction of finite simple groups is the topic of a review article written by R. W. Carter in 1965. Carter covers not only the original work by Chevalley, but also the later extensions of the theory.
by R. Steinberg, J. L. Tits and R. Ree [19].

J. R. Derome and W. T. Sharp showed how to define the n-j coefficients for any compact group in 1965, and they derived a number of their algebraic properties [20]. The letter 'j' in the name 'n-j coefficient' stands for 'module.' Since all the n-j coefficients can be defined in terms of the 3-j coefficients, the basic computational problem is to calculate the latter.

While Derome and Sharp present no specific algorithm for obtaining the 3-j coefficients, it is not hard to describe such a procedure. The calculation of the 3-j symbols is mathematically equivalent to constructing a basis for the trivial submodule of the tensor product of three irreducible modules. The trivial submodule of a reducible module is by definition the set of all vectors in the module which are annihilated by every element of the Lie algebra. As we shall see later, the determination of the trivial submodule amounts to solving a system of homogeneous linear Diophantine equations.

The problem of finding integer solutions of homogeneous linear equations comes up over and over again in Lie algebraic problems. One can view the process of solving such equations as that of constructing a basis for a finitely generated abelian group subject to a finite number of relations [21]. A simple ALGOL procedure for doing this was written by D. A. Smith in 1966.

The proof of the existence of Chevalley bases for Lie modules was put into a particularly elegant form by B. Kostant in 1966. We briefly explain some of the ideas here. Consider a simple complex Lie algebra with N positive roots, \( \alpha_1, \alpha_2, \ldots, \alpha_N \). For any sequence \( S = (s_1, s_2, \ldots, s_N) \) of non-negative integers, define the 'raising' and 'lowering' elements,
belonging to the universal enveloping algebra of the Lie algebra. If \( v \) is a nonzero vector of highest weight for an irreducible module, then the set of all the vectors \( f_s v \) spans the module. All but a finite number of these vectors are zero, and the nonzero ones need not be linearly independent. From these elements, however, one can extract a basis with respect to which the module action can be expressed in terms of integer matrices [22].

To illustrate these ideas, we turn again to the Lie algebra \( A_1 \). In the universal enveloping algebra, we introduce the divided powers

\[
\begin{align*}
\eta_k &= \frac{e_k}{k!} , \\
\zeta_k &= \frac{f_k}{k!} .
\end{align*}
\]

The highest weight of an irreducible module over \( A_1 \) is characterized by a single Dynkin index \( n \), which can be 0,1,2,\ldots. The dimension of the module is \( n+1 \). If \( v \) is a nonzero vector of highest weight, \( hv = nv \), then the vectors \( v, f_v, f_2 v, \ldots, f_n v \) form a basis for the module. The matrices of the basis vectors \( e, f, h \) of the Lie algebra \( A_1 \) with respect to this basis for the module are as follows:

\[
\begin{align*}
\eta_s &= \frac{(x_{\alpha_1})^{s_1}}{s_1!} \frac{(x_{\alpha_2})^{s_2}}{s_2!} \cdots \frac{(x_{\alpha_N})^{s_N}}{s_N!} , \\
\zeta_s &= \frac{(x_{-\alpha_1})^{s_1}}{s_1!} \frac{(x_{-\alpha_2})^{s_2}}{s_2!} \cdots \frac{(x_{-\alpha_N})^{s_N}}{s_N!} .
\end{align*}
\]
### Integer Clebsch-Gordan Coefficients

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>n</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>n-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>n-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
One should contrast these simple formulas for the action of the Lie algebra \( A_L \) on a module with the square-root laden formulas traditionally employed in angular momentum theory. In the quantum theory of angular momentum, one writes

\[
j = \frac{n}{2}, \quad m = \frac{n}{2} - k,
\]

and one uses the basis vectors

\[
|j m\rangle = \sqrt{(j - m)! (j + m)!} f_{j-m} v.
\]

From this one sees that all square roots in angular momentum theory can be removed simply by changing the normalization of the basis vectors.

In setting up the Chevalley basis for a simple Lie algebra, one needs to have a prescription for choosing the signs of the coefficients \( N_{\alpha, \beta} \). These signs are to some extent arbitrary, but they must satisfy

\[
N_{\beta, \alpha} = -N_{\alpha, \beta} = N_{-\alpha, -\beta}
\]

and they must be chosen in a manner consistent with the Jacobi identity for the Lie algebra \([23]\). An algorithm for making such a consistent choice of signs in a Chevalley basis was described by J. L. Tits in 1966. To explain this algorithm, we need to introduce some terminology. Let us define an addable pair of roots \((\alpha, \beta)\) to be a pair of nonzero roots \(\alpha\) and \(\beta\) such that \(\alpha + \beta\) is also a nonzero root. An addable pair \((\alpha, \beta)\) is called a special pair if \(\alpha\) and \(\beta - \alpha\) are positive. A special pair \((\alpha, \beta)\) is extraspecial if every special pair \((\gamma, \delta)\) with \(\alpha + \beta = \gamma + \delta\) satisfies \(\alpha \leq \gamma\) and hence also \(\delta \leq \beta\). The signs in a Chevalley basis can be chosen arbitrarily for the extraspecial pairs. The Jacobi identity is used to extend the sign prescription to the special pairs, and finally the sign for any addable pair is obtained from a special pair by the triangle rule: if \(\alpha + \beta + \gamma = 0\), then \(N_{\alpha, \beta}, N_{\beta, \gamma}\) and \(N_{\gamma, \alpha}\) all have the same sign.
In the mid-1960's physicists began to set up computer programs tackling individual Lie algebras on a piecemeal basis. Particular attention was directed to the Lie algebra of the group SU(3), partly because it is the next more complicated case after SU(2), and partly because it has important applications to the collective model of the atomic nucleus, as well as to elementary particle physics. For example, M. Herttua and P. Jauho in 1966 reported on an ALGOL program to compute the Clebsch-Gordan series for SU(3). In their work, the Clebsch-Gordan coefficients were not discussed [24].

While setting up a computer program for SU(2) Racah coefficients, J. Stein in 1967 discovered a binary algorithm for obtaining the greatest common divisor of two integers [25]. Any greatest common divisor algorithm can be used to speed up the solution of linear Diophantine equations. The binary algorithm finds the greatest common divisor using shifting, parity testing, and subtraction. The algorithm is fast because, unlike Euclid's algorithm, it requires no divisions. D. E. Knuth (The Art of Computer Programming, vol. 2, section 4.5.2) says that this algorithm was actually discovered earlier by R. L. Silver and J. Terzian.

J. L. Tits in 1967 published tables providing some basic information concerning simple Lie groups and their representations [26].

L. C. Biedenharn and various collaborators embarked on an ambitious program to compute the Clebsch-Gordan and Racah coefficients for the unitary groups [27]. They did not use Chevalley bases, but instead used the subgroup chain \( U(n) \supset U(n-1) \supset \ldots \supset U(2) \), following the 1950 Gelfand-Zetlin paper. In 1967 they proposed a canonical definition for the coupling coefficients of the unitary groups, based on an embedding of the irreducible representations of \( U(n) \) into the totally...
symmetric irreducible representations of $U(n^2)$.

In the following year, K. B. Wolf announced a set of FORTRAN subroutines for handling polynomial bases for representations of the group $U(n)$, using the same subgroup-chain strategy [28].

V. K. Agrawala and J. G. F. Belinfante in 1968 developed a graphical recoupling theory for compact groups, based on the algebraic formalism of J. R. Derome and W. T. Sharp. Lines are used to represent modules, and nodes represent module homomorphisms. Parallel lines represent tensor products, while crossed lines represent exchange operators [29]. For example, the graph

\[
\begin{array}{c}
B \\
\text{A} \\
\text{A} \\
\text{B}
\end{array}
\]

represents the operator taking the vector $u \otimes v$ in the module $A \otimes B$ into the vector $v \otimes u$ in the module $B \otimes A$.

Using this notation, one can distinguish two kinds of 3-j symbols:

\[
\begin{array}{c}
A \\
B \quad \text{-----} \\
C
\end{array} \quad \begin{array}{c}
A \\
\text{-----} \quad B \\
C
\end{array}
\]

Here, the solid lines represent irreducible modules $A$, $B$, and $C$, while the dotted lines represent the trivial submodule of their tensor product. The graph on the left represents the inclusion mapping, while the graph on the right represents the module homomorphism which projects the tensor product module into its trivial submodule. This is unique because the trivial submodule is an isotypical component. This projection can be constructed
explicitly by averaging over the group, using the Haar integral.

All other n-j symbols can be defined in terms of the 3-j symbols. For example, the 9-j symbol is defined by the following picture.

The calculation of 3-j symbols using Chevalley bases and Diophantine linear equations was illustrated for the simple Lie algebra $A_2$ in 1969 by J. G. F. Belinfante and B. Kolman in the third of a series of survey articles on Lie algebras and their representations [30].

To explain this method, we consider here a simpler problem, the computation of the 3-j coefficients for spinor-spinor-vector coupling in the simple Lie algebra $A_1$. The two spinor modules both have Dynkin index $n = 1$ (spin $j = \frac{1}{2}$), while the vector module has Dynkin index $n = 2$ (spin $j = 1$). If $v$ and $v'$ are highest weight vectors for the spinor modules, and $v''$ a highest weight vector for the vector module, then a basis for the tensor product of these three irreducible modules is given...
by the vectors

\[ v_{abc} = f_a \otimes f_b \otimes f_c \]

where \( a, b = 0, 1 \) and \( c = 0, 1, 2 \). As before, the elements \( f_n = f^n/n! \) are divided powers in the universal enveloping algebra.

The problem is to find the trivial submodule, consisting of vectors

\[ t = \sum_{a,b,c} t_{abc} v_{abc} \]

satisfying \( et = ft = ht = 0 \). The numerical coefficients \( t_{abc} \) are the 3-j coefficients. The equation \( ht = 0 \) implies \( t_{abc} = 0 \) unless \( a + b + c = 2 \). The equation \( ft = 0 \) yields a system of homogeneous equations,

\[
\begin{align*}
0 &= t_{002} + 2 t_{101} \\
0 &= t_{002} + 2 t_{011} \\
0 &= t_{101} + t_{011} + t_{110}.
\end{align*}
\]

The remaining condition \( et = 0 \) gives nothing new. The solution of the Diophantine system is

\[ t_{110} = -2 t_{101} = -2 t_{011} = t_{002}, \]

and therefore

\[ 2 v_{110} - v_{101} - v_{011} + 2 v_{002} \]

is a basis for the trivial submodule. We see that the trivial submodule in this case happens to be one-dimensional. This holds generally for the Lie algebra \( A_2 \), but not for other simple Lie algebras.

Extensive calculations of the characters and related basic information for irreducible representations of simple Lie algebras were begun in 1968 by V. K. Agrawala and J. G. F. Belinfante on a UNIVAC 1108 machine using FORTRAN V. Up to that time, such computations reported in the literature had been done by hand and
were mostly limited to algebras of rank three or less. The program uses the Dynkin layer algorithm to obtain weight systems, and the Freudenthal formula for weight multiplicities, making no use of Weyl reflections. The FORTRAN program has subroutines to automatically scan through the simple Lie algebras of all types with rank up to eight, and to examine those irreducible modules found to have dimension less than a thousand. The dimensions of the modules and the values of the second order Casimir operator are found from Weyl's formula. There are also subroutines to print out useful information about the duals of the modules, and the Young tableaux corresponding to Dynkin indices. Originally, an integer-mode version of the Gauss-Jordan reduction algorithm was used to invert the Cartan matrices, but this caused overflow problems for the C-type algebras of ranks 7 and 8, and for the D-type algebras of ranks 6, 7 and 8. No overflow occurs for A-type algebras up to rank 20. Perhaps this overflow problem could be eliminated by renumbering the simple roots. In the shortened ALGOL version of these programs published in 1969, the problems associated with the Cartan matrix inversion were by-passed by using empirical regularities found in Dynkin's tables [31].

4. Recent Work, Since 1970

A question raised in the paper by Agrawala and Belinfante is whether the Freudenthal algorithm is actually the most efficient method for computing characters. For simplicity they had used the Freudenthal formula to find the multiplicity of every weight, disregarding simplifications possible by using Weyl reflections. M. I. Krusemeyer in 1971 published an improved ALGOL 60 procedure in which, by analogy with most hand computations, the Freudenthal formula is used to compute only the multiplicities of the dominant weights. Other weights encountered in the course of the calculation are transformed into dominant weights by Weyl reflections [32].
Besides the Freudenthal algorithm for computing characters of Lie modules, there are several alternative methods which involve the Weyl group. R. E. Beck and B. Kolman in 1971 constructed a program to generate the Weyl group on a computer, paving the way for a comparison of the various algorithms [33]. Their procedure is to first generate a subgroup isomorphic to a symmetric group $S_n$, and then to generate the Weyl group by coset enumeration, using an explicit presentation of the Weyl group. Each element of the Weyl group is stored as a word, using the simple Weyl reflections as alphabet. The large size of the Weyl group causes storage problems, limiting the methods to rank four. More compact storage is possible by breaking each word into syllables and storing only the syllables [40].

Consider for example the Weyl group of the simple Lie algebra $A_3$. Let 0 denote the identity element, and let 1, 2 and 3 denote the Weyl reflections corresponding to the first, second and third simple roots, respectively. The Weyl group has 24 elements, and the longest word is 121321, which has six letters. So one could require $6 \times 24 = 144$ storage locations to store the words directly as an array. Each word however can be constructed out of three syllables. The first syllable is either 0 or 1, the second is 0, 2 or 21, and the third is 0, 3, 32 or 321. The storage needed to contain these syllables as an array is only $3 \times 4 \times 3 = 36$ locations. For higher rank Lie algebras, the savings are even better.

It is well-known that by employing higher-order and polarized Casimir invariants, one can compute 6-j symbols directly without first setting up the 3-j symbols. V. K. Agrawala and J. G. F. Belinfante in 1971 published ALGOL procedures for the relevant Casimir invariants for $SU(n)$ representations, obtained via their diagram calculus [34].

N. Burgoyne in 1971 reported the first successful computer
implementation of algorithms involving Chevalley bases for Lie modules [35]. For each weight \( \nu \) of an irreducible module with highest weight \( \lambda \) he defines a matrix whose entries are integers \( a_{ST} \) defined by

\[
e^S e^T v = a_{ST} v
\]

Here, as before, \( S = (s_1, \ldots, s_N) \) and \( T = (t_1, t_2, \ldots, t_N) \) are sequences of non-negative integers satisfying

\[
\sum_{k=1}^{N} s_k \alpha_k = \sum_{k=1}^{N} t_k \alpha_k = \lambda - \mu,
\]

and \( v \) is a nonzero vector of highest weight: \( hv = \lambda(h) v \). The matrices \( a_{ST} \) can be computed directly from the commutation relations of the Lie algebra. To compute a Chevalley basis for a Lie module, one needs to know all linear relations satisfied by the vectors \( e^T v \). If

\[
\sum_{T} r_T f_T v = 0
\]

is such a relation, then applying \( e_S \), we obtain

\[
\sum_{T} a_{ST} r_T = 0.
\]

So, all these relations can be computed by finding the null space of the matrix \( a_{ST} \).

In another paper, N. Burgoyne and C. Williamson computed the elementary divisors of the matrices \( a_{ST} \) to obtain results on the multiplicities of weights in the characteristic \( p \) case [36]. They remark that their algorithm for setting up the matrix \( a_{ST} \) has a long execution time. These programs were written in an assembly language for an IBM 360-50 machine.

To speed up the solution of systems of Diophantine linear equations, one needs not only to be able to calculate the greatest common divisor \( d \) of a list of integers \( a_1, a_2, \ldots, a_n \), but one also needs to be able to produce a set of multipliers \( x_1, x_2, \ldots, x_n \) such that \( d = a_1 x_1 + a_2 x_2 + \ldots + a_n x_n \). G. H. Bradley in 1972 published
an ALGOL procedure for this, which is based on a modification of Euclid's algorithm [37]. One can gain a factor two in efficiency by allowing negative remainders in Euclid's algorithm.

D. N. Verma in 1971 used a computer to study a conjecture concerning a certain harmonic polynomial related to Weyl's dimension formula [38]. This conjecture was settled in 1974 by S. G. Hulsurkar [51].

R. E. Beck and B. Kolman in 1972 published a survey of their further experience using computers to study representations of Lie algebras [39]. They also reported on yet another variant of Freudenthal's algorithm in which one again computes only the multiplicities of the dominant weights, as in Krusemeyer's version [41]. The difference is now that one does not keep transforming every weight into a dominant weight, but instead one runs through the entire weight system just once, applying only a single Weyl reflection on each of the non-dominant weights, transforming them into higher weights with known multiplicities [47].

During this same period, J. G. F. Belinfante and B. Kolman published a monograph surveying the present status of the applied theory of Lie groups, Lie algebras, and their representations. This monograph concentrates on presenting the results needed to understand the current applied literature, and it includes an account of some experience using computers [42].

Readable accounts of Chevalley bases can be found in the recent book on Lie algebras by J. E. Humphreys and in the book on Chevalley groups by R. W. Carter, both published in 1972 [43, 44].
R. E. Beck and B. Kolman completed their comparison of algorithms for inner and outer multiplicities in 1973. They found that for inner multiplicities, the Freudenthal algorithm is best [45]. Methods requiring the generation of the full Weyl group are less efficient because the Weyl group grows rapidly in size with increasing rank. For outer multiplicities (Clebsch-Gordan series), they report that a formula of G. Racah provides the shortest computation times [46]. Although Racah's formula appears to involve the Weyl group, one need not actually compute the full Weyl group because many terms in Racah's formula usually drop out for low-dimensional representations. By using only some properties of the Weyl group, and an appropriate stopping condition, computer programs using Racah's formula are quite fast [48,50].

Any module over a Lie algebra can by restriction also be considered as a module over any subalgebra. An irreducible module over a simple Lie algebra will generally be reducible upon restriction to a semisimple subalgebra, and its decomposition into irreducible submodules over the subalgebra is known as a branching rule. Branching rules are important when one uses the subgroup-chain approach to the coupling coefficients. Extensive computer-generated tables of branching rules were published by J. Patera and D. Sankoff in 1973. This work also includes a list of all irreducible modules of dimension less than a thousand over any simple Lie algebra of rank up to eight [49].

Many ideas used in the algorithms we have been discussing first arose in the study of Chevalley's finite simple groups. As one might expect, the theory of Chevalley groups has itself continued to be developed over the past twenty years. We mention here just one recent development in this field, the publication by H. Behr in 1975 of explicit presentations for these groups [52].
5. **Outlook**

We have surveyed the literature on algorithms concerning the representations of simple Lie algebras, with special reference to the problem of computing the 3-j coefficients using Chevalley bases. Since many of the relevant algorithms were originally developed in connection with hand calculations, only informal verbal descriptions of them were given in the literature. The availability of computing machines has led to renewed interest in the development of uniform methods for studying representations of Lie groups. For computer use, the classical algorithms must first be translated into precise and explicit programs. Publication of such formal computer programs is necessary so they can be verified, and so that meaningful comparisons of running times can be made when there are competing algorithms which accomplish the same task.

**Bibliography (arranged in chronological order)**


*School of Mathematics*  
*Georgia Institute of Technology*  
*Atlanta, Georgia 30332*