Lithuanian Journal of Physics, Vol. **44**, No. 2, pp. 85–89 (2004) doi:10.3952/lithjphys.44201

A TRIBUTE TO PROFESSOR JUCYS: ON THE INFLUENCE OF HIS THEORIES ON THE MODERN ATOMIC PHYSICS COMPUTING PROGRAMS

M. Klapisch *

ARTEP Inc., Ellicott City, MD 21042, USA

Received 17 June 2004

Dedicated to the 100th anniversary of Professor A. Jucys

The impetus given by the work of A. Jucys, combining creativity and mathematical rigour, continues to thrive and to find very useful applications. Some earlier implementations of the graphical theory of Jucys, Levinson, and Vanagas as well as its future developments and applications are shortly characterized.

Keywords: angular momentum graphs, $3nj$ coefficients, atomic physics codes, atomic spectra

PACS: 31.15.–p, 32.30.–r

1. Introduction. Historical notes

Some contributions to physics are so fundamental that they open an entirely new way of looking at things. So is the work of Professor Jucys and coworkers, and specifically the famous YLV [1], published in 1962. It is an abstract, rather than mathematical work, but after more than 40 years it continues to be cited and to generate enthusiasm and new algorithms. I found 94 citations in the Science Citation Index, the most recent ones from 2003, and they cover a rather broad spectrum of research interests. Unfortunately, some more advanced work [2–5] published later by him and his former students was written in Russian and I could not have access to them. It is remarkable that Professor Jucys (this is the way I shall write his name from now on) was able, with limited resources, to start a research group that continues vibrantly to this day to innovate both on the purely theoretical aspect and on the computational aspect of atomic theory. I would dare to say that there is no modern computer program for atomic physics – i. e. necessarily computing recoupling coefficients – that is not based, one way or another, on the ground breaking work of Jucys, Levinson, Vanagas, and others.

I had the privilege to meet Professor Jucys in 1969 when I was preparing my PhD at Laboratoire Aime Cotton (LAC), in Orsay, France. This laboratory was then primarily concerned with experimental atomic spectroscopy. A small theory group (which included J. Bauche, C. Bauche-Arnoult, Y. Bordarier, S. Feneuille, and myself) had been started in 1964 at the initiative of the director, late Professor L. Jacquinot. Professor B. Judd was then invited to stay for two years to introduce us to the new developments in atomic theory. Professor Jucys was invited in November 1969 by C. Moser, of the Centre Européen de Calcul Atomique et Moléculaire, Orsay, France (CECAM), but he actually came to LAC and gave three seminars. He had then the opportunity to explain to us the mathematical foundations of his graphical theory. At first sight, he gave the impression of a small old man, but when he was lecturing, his mastery of the subject, and his rigour in demonstration were very powerful. We were very impressed because we had had previously an introduction to graphical representation in the lectures by B. Judd, but in Judd's book [6], there are no arrows nor node signs, so the phases cannot be computed, and it did not lead to practical applications. It was, however, useful for finding symmetries and selection rules.

It was already understood some time before that that the study of complex spectra would require computer programs, and it became clear after Professor Jucys' visit that the graphical theory would be a requisite tool for this purpose. In July 1970, after working intensively

* Contractor to The Naval Research Laboratory, Washington, DC 20375, USA

ISSN 1648-8504

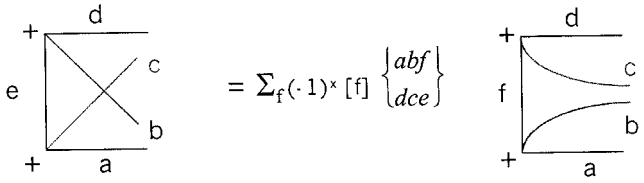

Fig. 1.

on YLV, Y. Bordarier defended his PhD thesis [7] on such a program. This program uses already the concept of choice of the best cut [8]. Professor Jucys was invited again, to be member of the jury for that occasion. That was really the start of the possibility of classification of complex spectra of heavy elements at LAC, and many papers were published using this code. Unfortunately, this program was never published.

At that time, I was more interested in the radial part of the wave functions. In the early 1980s, when I became a professor at the Hebrew University of Jerusalem, I got involved in electron–atom collisions for collisional radiative models for atoms in plasmas. I tried then to use available computer programs [9] for this purpose from the Computer Physics Communication Library. It turned out that the computation of the recoupling coefficients, using a code of P.G. Burke [10] (NJSYM) was extremely slow. In fact, the computations were not feasible with the computer we had at that time.

So we had no choice but to write our own program. In the meantime, a book by Lindgren and Morrisson [11] had been published, with extensive tutorial on use of graphical methods for atomic physics. However, upon examination, their notations were found not to be convenient for coding, because they introduced different kinds of lines and arrows. So we turned again to YLV [1], and that was the basis for our code NJGRAF [12], which was written in such a way that it could easily replace NJSYM. The former turned out to be orders of magnitude quicker than the latter. It is instructive to understand where this difference of performance comes from.

2. Earlier implementation of the graphical theory

Thanks to graphical representation it is easy to see what is the difference. In NJSYM, the only process taken into account was the interchange of two lines, shown in Fig. 1. Each interchange introduces a summation variable. For the sake of simplicity, arrows are omitted. With this process, one can always resolve an angular momentum recoupling coefficient into a sum

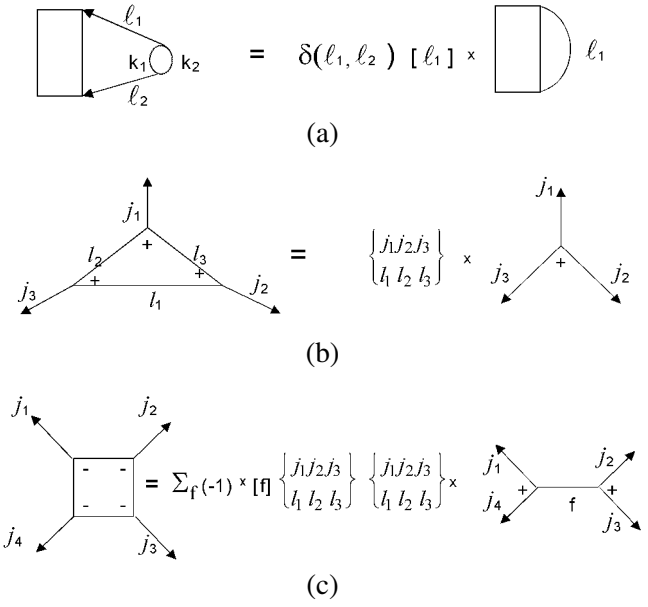

Fig. 2.

over product of $6j$ symbols, but the decomposition will generally not be optimal. An example of nonoptimal decomposition is finding the left-hand side of Eq. (1) instead of its right-hand side. This is the well known Biedenharn–Elliott sum rule:

$$\sum_x [x](-1)^x \left\{ \begin{matrix} a\,b\,x \\ c\,d\,p \end{matrix} \right\} \left\{ \begin{matrix} c\,d\,x \\ e\,f\,q \end{matrix} \right\} \left\{ \begin{matrix} e\,f\,x \\ b\,a\,r \end{matrix} \right\}$$

$$= (-1)^s \left\{ \begin{matrix} p\,q\,r \\ e\,a\,d \end{matrix} \right\} \left\{ \begin{matrix} p\,q\,r \\ f\,b\,c \end{matrix} \right\}, \qquad (1)$$

where S is the sum of all angular momenta except x. It is obvious that the right-hand side is much quicker to compute than the left-hand side. The time ratio will depend on the number of values that the summation variable x can take. This range depends on the values of the angular momenta through triangular inequalities.

Also it is impossible, with this restricted reduction scheme, to take into account the fact that some angular momenta have zero value, thus losing the possibility of great simplifications. Once this shortcoming was realized, we decided to make full use of the graphical method of analysis for evaluating all the recoupling coefficients and matrix elements of tensor operators that are needed for the various atomic processes in plasmas: energies, radiative transitions, auto-ionization, and electron collision excitation and ionization.

In our program NJGRAF [12], we endeavoured to identify loops of different orders, so as to use the well-known rules appearing in YLV, as described in Fig. 2.

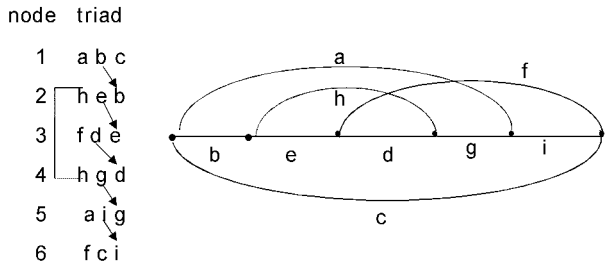

Fig. 3.

The aim of most codes is to transform a recoupling coefficient into a sum of products of $6j$s. Some codes also recognize $9j$s and even higher njs, but since simple closed formulas for them are not available, it is not necessarily a practical advantage.

One question is how does one store graphs in the computer. Several methods have been described in the literature [7, 13] for that purpose. For instance, one can store the nodes, their sign, and the lines coming into them with their arrows. It usually turns out that it is useful to also list the lines and the nodes they are connected with. The memory storage is nowadays not usually a problem.

The identification of the various loops is also not too difficult. It is sufficient to follow the lines through the nodes, until one comes back to the departing node. The more difficult question is to find the optimal decomposition – i. e. according to the criterion of Jucys [8], the one that will have the least summation variables. The methods for achieving this goal will be described in the next section. In NJGRAF these methods were not implemented. Indeed, we realized that most of the recoupling coefficients occurring normally in the atomic processes we consider – e. g., energies or transitions – were quite simple, and on the other hand, the research of the optimal cut could be time consuming. Consequently, we scanned the graph for loops of order 2, 3, and 4, and we were content with cutting along the lines of the first loop of each kind encountered in the scan. As mentioned above, this procedure was already very much quicker than NJSYM. The graphs are stored in the computer in the so-called flat diagram representation, inspired by the book of Brink and Satchler [14].

An example is shown in Fig. 3.

One sees the principle on the left side of Fig. 3: the main axis goes from the second to the third position of the momenta in the triad. If a momentum appears two times in the first position in nodes n_1 and n_2, e. g., as above in nodes 2 and 4, then there is a loop of order $n_2 - n_1 + 1$, here $4 - 2 + 1 = 3$. One then scans the graph for loop of order 2, cuts them out, then loops of order 3, etc. After each loop of order k is cut, the scan starts again for loops of order $k-1$. The node signs and arrows are not indicated here for simplicity, but they are taken into account in the code. No attempt is made to find the optimal cuts. Care has to be taken to ensure one finds loops involving the end nodes, but all in all the procedure is very quick.

It should be noted that the original program NJGRAF had some mistakes, that were noticed by some users [15] and other authors [13]. These errors were corrected, and the corrected version, although never published, was used in several packages [9, 16] as well as in the collisional radiative model HULLAC [17]. These packages were (and still are) used in a large number of computations, and thus it can be said that the graphical theory in its first application to atomic codes was extremely fruitful.

3. Further developments

The HULLAC code [17] is organized in such a way that all the angular coefficients for the different processes are computed at the beginning, the same formulas being used many times, and this part of the computation takes a negligible amount of time, compared to, e. g., collision cross-sections. However, this is not necessarily the case in other packages [18], where the number of configuration state functions (CSF) is very large, and can grow dynamically, and the angular computations are an important part of the computing time.

Thus, the question arises of how to find the optimal decomposition, that is, as mentioned above, the one that will give the least summation variables.

This involves, for each order k of loops (or cuts) sought, two stages:

(i) Being able to find and store all the loops of order k.
(ii) Being able to judge which one is the "best".

The last point is a little ambiguous, since the number of summation variables will be different if we consider only $6j$s in the final formula, or if we also have closed formulas for, e. g., $9j$s, $12j$s, etc. The first point is described in the work of Rumšas and Jucys [19] and it involves building the matrix of incidence of the graph. This is a rectangular matrix, the lines of which correspond to the nodes, and the columns to the branches. The elements $r(i, j) = 1$ if the node i is an extremity of line j, and 0 otherwise. Then considering the logical sum of k lines of the matrix gives the number of

branches that are connected in a loop of order k. This is the equivalent of scanning the flat diagram as mentioned above.

The second point is more complicated, and involves two more matrices, the matrix of the common branches, and the matrix associated with the graph, which allows one to evaluate the "distance" between any two nodes. If one decides to end up with higher njs than $6j$s, then there is another choice to be made, since, e. g., there are two species of $12j$s, etc.

This process is quite elaborate, and it is not always worth the effort. That was the rationale for skipping it in NJGRAF.

The Belgian group of Fack et al. [20, 21] used another approach for finding all the relevant cycles, based on an algorithm of Vismara [22]. Then, they first get rid of all bubbles (2-loop) and triangles (3-loop). For the rest, they define a cost of each branch as the difference in length of the two smallest loops in which this branch participate. The cost of the loop is then the minimum cost of its branches. Then, after computing the cost of all the girth loops – i. e. the loops with the smallest order k – they select the one with minimal cost. They devised a method for choosing the best of these loops if there are several with the same minimum cost. With this algorithm, they achieve a significant gain in computing time for complicated graph.

On the other hand, the Lithuanian school [23, 24] took a more mathematical approach, and went one step further in the analysis, by considering graphs that are based on second quantization. Quasispin [25] is also considered explicitly. They were able to tabulate reduced matrix elements of any two-particle operators, depending on the subshells on which they act, but independent of the occupation number of the subshells. The remaining graph gets then much simpler. The usual rules are again used to reduce the graph in a sum of products of $6j$s. The advantages of the latter approach are:

(i) These tabulated universal quantities allow great simplifications of the graphs, and thus the computations are several times quicker.

(ii) The introduction of quasispin gives the possibility of evaluating the coefficients of fractional parentage, so, e. g., the relativistic computations are no more limited to two electrons on the $j = 9/2$ shell.

4. Conclusion

The different codes mentioned above all are based on the breakthrough of Professor Jucys. Some codes [12] just use the simple graphical methods described in YLV. Some use the algorithm of the best cut, either based on graph theory [7], or on a heuristic approach [20]. In even more advanced methods, the followers of Professor Jucys in Vilnius use second quantification and quasispin to generate even more efficient codes [23–25]. All this shows that the impetus given by Professor Jucys, combining creativity and mathematical rigour, continues to thrive and to find very useful applications. All modern atomic physics codes use Professor Jucys' methods one way or another.

Acknowledgements

This work was supported by the USDOE through a contract with the Laser Plasma Branch, Naval Research Laboratory. I also acknowledge the contribution of A. Bar-Shalom, who wrote most of NJGRAF, and the help of J. Oreg.

References

[1] A.P. Jucys, I.B. Levinson, and V.V. Vanagas, *Mathematical Apparatus of the Theory of Angular Momentum* (Israel Program for Scientific Translations, Jerusalem, 1962).

[2] A.P. Jucys and A.J. Savukynas, *Mathematical Foundations of the Atomic Theory* (Mokslas, Vilnius, 1973) [in Russian].

[3] A.P. Jucys and A.A. Bandzaitis, *Theory of Angular Momentum in Quantum Mechanics* (Mokslas, Vilnius, 1977) [in Russian].

[4] Z. Rudzikas and J.M. Kaniauskas, *Quasispin and Isospin in the Theory of Atoms* (Mokslas, Vilnius, 1984) [in Russian].

[5] G. Gaigalas, J.M. Kaniauskas, and Z. Rudzikas, Liet. Fiz. Rink. **25**, 3 (1985) [Sov. Phys. – Collection].

[6] B.R. Judd, *Operator Techniques in Atomic Spectroscopy* (McGraw-Hill, New York, 1963).

[7] Y. Bordarier, Nouvelle méthode pour calculer les coefficients angulaires, Ph.D. thesis, Université Paris-Orsay, (1970).

[8] P.D. Rumšas, A.A. Bandzaitis, and A.P. Jucys, Liet. Fiz. Rink. **5**, 197 (1965).

[9] K.A. Berrington, W.B. Eissner, and P.H. Norrington, Comput. Phys. Commun. **92**(2), 290 (1995).

[10] P.G. Burke, Comput. Phys. Commun. **1**(4), 241 (1970).

[11] I. Lindgren and J. Morrison, *Atomic Many-Body Theory* (Springer-Verlag, Berlin, 1982).

[12] A. Bar-Shalom and M. Klapisch, Comput. Phys. Commun. **50**, 375 (1988).
[13] V. Fack, S.N. Pitre, and J. Van der Jeugt, Comput. Phys. Commun. **83**, 275 (1994).
[14] D.M. Brink and G.R. Satchler, *Angular Momentum* (Clarendon Press, Oxford, 1968).
[15] I.P. Grant and C. Froese Fischer (private communications).
[16] F.A. Parpia, C. Froese-Fischer, and I.P. Grant, Comput. Phys. Commun. **94**(2), 249 (1996).
[17] A. Bar-Shalom, M. Klapisch, and J. Oreg, J. Quant. Spectros. Radiat. Transfer **71**, 169 (2001).
[18] S. Fritzsche, C. Froese-Fischer, and G. Gaigalas, Comput. Phys. Commun. **148**(1), 103 (2002).
[19] P.D. Rumšas and A.P. Jucys, Liet. Fiz. Rink. **5**, 185 (1965).
[20] D. Van Dyck and V. Fack, Comput. Phys. Commun. **154**(3), 219 (2003).
[21] D. Van Dyck and V. Fack, Comput. Phys. Commun. **151**(3), 354 (2003).
[22] P. Vismara, Electronic J. Combin. **4**(1), R9 (1997).
[23] G. Gaigalas, S. Fritzsche, and I.P. Grant, Comput. Phys. Commun. **139**(3), 263 (2001).
[24] G. Gaigalas and S. Fritzsche, Comput. Phys. Commun. **148**(3), 349 (2002).
[25] G. Gaigalas, Z. Rudzikas, and C. Froese-Fischer, J. Phys. B **30**(17), 3747 (1997).

PROFESORIAUS A. JUCIO ATMINIMUI: APIE JO TEORIJŲ ĮTAKĄ ŠIUOLAIKINĖMS ATOMO FIZIKOS SKAIČIAVIMŲ PROGRAMOMS

M. Klapisch

ARTEP, Ellicot City, JAV

Santrauka

Trumpai apžvelgiama prof. A. Jucio darbų, ypač jo ir mokinių išplėtoto grafinio metodo, įtaka tolesnei atomo fizikos skaičiavimo metodų raidai. Nagrinėjami tiek ankstyvieji grafinio metodo taikymai, tiek vėlesnė raida, išplėtimas į antrinio kvantavimo vaizdavimą. Apibūdintos kai kurios bendros atominių dydžių skaičiavimo programos.