Dyes and Pigments 5 (1984) 165–170

Colour Prediction of Organic Dye Molecules on a Microcomputer—a Readily Available Form of the PPP–MO Method for Teaching and Research

Giorgio F. Castellani and Jan C. J. Bart†

Istituto di Chimica Industriale, Universita di Messina, 98100 Messina, Italy

(Received 12 November, 1982)

SUMMARY

The Pariser–Parr–Pople molecular orbital (PPP–MO) method, which provides a way of calculating electronic absorption spectroscopic properties of organic dye molecules, such as the position of the absorption maximum in the visible spectrum, its intensity and polarisation direction, has been implemented in PASCAL language on a 64 kB Apple II computer The system consists of two modes (1) Graphic mode and (2) PPP mode, which are controlled separately or successively The system occupies up to 32 kB of memory, needs 1 minifloppy disk in the entire execution of the program, and handles up to 30 atoms routinely with configuration interaction

1. INTRODUCTION

Colour and constitution studies of organic molecules have received great benefit from the quantum theory. The quantitative prediction of light absorption has been aided by the development of molecular orbital theory and the advent of modern computing facilities. In this respect, the Pariser–Parr–Pople (PPP) MO method[1–3] in particular has shown its usefulness in the last three decades due to its ease of use, predictive reliability and small demand on computer capacity. In fact, it has recently

---

† Address for correspondence: CNR, Via Salita S Lucia sopra Contesse 39, 98103 Pistunina, Messina, Italy.

Dyes and Pigments 0143-7208/84/$03 00 Printed in Great Britain

been stated that the PPP–MO method (with a mathematical formalism similar to that of the Huckel MO method[4]) has no rival in industrial research [5] Advantages and problems connected with the method have been discussed by Griffiths[5] and others [6,7] In order to increase further the value of the technique we have now revised and adapted a widespread PPP–MO computer program to run on a low-cost self-contained microcomputer In combination with the recently published generalised PPP–MO parameters,[5] this program version is expected to increase awareness of the value of the technique for routine application.

2 SYSTEM SOFTWARE AND HARDWARE

In its original form the PPP–MO program, the subject of this paper, was implemented on the ICL-1906A computer of the University of Leeds, where it has been used routinely for colour prediction of manifold dye classes [5 8] These predictions include the calculation of (a) the wavelength of maximum absorption (λ_{max}) of the visible band (approximate indication of the hue of the dye), (b) the absorption band intensity (at λ_{max}, estimate of the effectiveness of a dye as a colouring agent); (c) the polarisation direction of the absorption band Input parameters include structural data for the π-electron part of the (usually planar) molecule (bond lengths and angles) together with electronic and energy parameters Whereas core charge values define the electronic parameters, the energy parameters are described by a valence state ionisation potential ($VSIP_n$) and repulsion energy (γ_n) for each atom and a resonance integral value (β_{mn}) for each pair of bonded atoms. Approaches to parameter derivation have been discussed extensively elsewhere[5] and will not be repeated here The electron repulsion energy between two electrons residing on different atoms is evaluated according to Nishimoto and Mataga.[9] With these data simple Huckel-type calculations are repeated to self-consistency, initially ignoring electron repulsion effects Ground and excited state energies are computed from the molecular orbital wave functions Restricted configuration interaction (CI) improves the calculated transition energies (i e generally taken as λ_{max} in non-polar solvents). Absorption intensities are given as oscillator strengths (f values) which are derived from the transition dipole moments, based on the molecular geometry and the form of the molecular orbital wave functions As pointed out elsewhere,[5] surprisingly little systematic attention has been paid so far to this aspect and the (industrially) important prediction of absorption intensities is generally less satisfactory.[10]

The widespread diffusion of personal computers and the high capabilities of their hardware and software suggest their possible use for quantum-chemical calculations, once primarily the domain of large computers To this purpose we have considered the use of an Apple II computer because of its wide availability, flexibility and extensive software facilities. The configuration of our system consists of: a 9 inch green monitor display, 2 built-in disk drives, a language card, and a PASCAL compiler, the 64 kB RAM includes a 14 kB operating system, a 2 kB memory-mapped I/O and over 40 kB user's space The capacity of one minifloppy disk is 140 kB.

The original PPP program was written in a language (ALGOL) not available on most microcomputers However, the various PASCAL implementations (UCSD, PASCAL, PASCAL/M, PASCAL/Z, etc.) conform to a grammar very close to ALGOL and translation was envisaged as a not unsurmountable problem. The main difficulty lay in the optimal use of the limited resources to accommodate the program with the available memory and to obtain acceptable execution times.

It is necessary at this point to stress the fact that PASCAL does not allow run-time dimensioning of arrays while ALGOL does This implies that in PASCAL the maximum number of array elements to be used must be determined at compilation time Initially we used dimensions of 25 × 25 for all arrays. This allows treatment of molecules with up to 25 atoms and the analysis of 25 transitions for CI. This corresponds to a memory occupation of 15 kB for the arrays.

The PASCAL implementation used was UCSD PASCAL though it is felt that this compiler is not the best for treatment of our problem, mainly for three reasons: (a) The compiler generates an intermediate code (the so-called P-code) which is then interpreted. This slows down the execution with respect to a direct machine-code; also, the available memory is affected due to the need for the presence of an interpreter program during execution. (b) Moreover, functioning of the compiler is poorly documented and the code generated could be shortened and memory utilisation optimised only by extensive trial-and-error. (c) Finally, the system lacks debug facilities, the error messages are cryptic and it is necessary to reinitialise the system when programming errors occur leading to loss of previous results and somtimes of hours of computing time However, this compiler was used as it is the only one available to us at the present time

To allow the program (15 kB of code) and its data area (16 kB) to fit in, memory segmentation is required In this case it is important to understand exactly what happens in the Apple memory at execution time Finally, we have reached a segmentation form that allows the use of a 30 × 30 dimension for all arrays allowing the study of molecules with 30 atoms and 30 transitions It was necessary to split some procedures into two or more due to the limitation to the procedure length in the Apple version of UCSD PASCAL (1 2 kB of code)

In the present configuration the execution time for the study of a typical 20-atom molecule (indigo), with 20 selected transitions, is about 4 h The major part of this time is devoted to diagonalisation procedures (15 min per cycle for a 20 × 20 matrix, 13 cycles) We used the Jacobi method, implemented by Cooper,[11] after revision of the published listing and further modifications to reduce the execution time by about 50% We believe that further optimisations are possible and are confident that we will be able to lower the execution time significantly in the near future

Comparison with the results of a test calculation on an ICL-1906A installation showed a slight decrease in precision of the calculations in the case of the microcomputer This leads to the need for more cycles in the SCF procedure to reach convergence (typically ten instead of seven cycles)

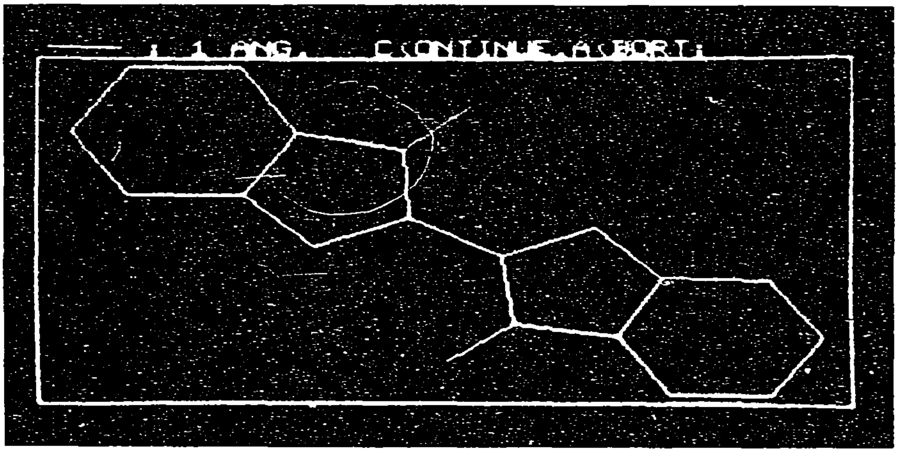

Fig. 1. Display of molecular geometry

and a loss in significance of the results beyond the second decimal digit This problem can probably be by-passed by the use of a better diagonalisation routine, such as the Householder inverse iteration method.[12,13]

Use of a CCS arithmetic processor to execute some functions (sin, cos, sqrt, sqr) speeds up the calculation time by some 10%, but causes an unacceptable loss in precision. This is due to truncation of the results by the processor

In order to check the input geometry before calculation of the electronic properties, we have written a separate plotting routine which uses the same input data as the PPP–MO program After display of the geometry on the monitor (execution time, 40 s; Fig. 1) the PPP program is started Even though this procedure requires recalculation of the geometry, this solution was preferred in view of the low computing time involved The graphical feature will be merged with the PPP–MO program in the future Copies of the program set are available from the authors at a small charge

The proposed system is not only a useful research tool, but can also readily find application in computer-assisted instruction for students

ACKNOWLEDGEMENTS

We would like to thank Dr J Griffiths and the Department of Colour Chemistry, University of Leeds, for their co-operation with this work

REFERENCES

1 R Pariser and R G Parr, *J Chem Phys*, **21**, 466 (1953)
2 R Pariser and R G Parr, *J Chem Phys*, **21**, 767 (1953)
3 J A Pople, *Trans Faraday Soc*, **49**, 1375 (1953)
4 R L Flurry, *Molecular orbital theories of bonding in organic molecules* London, E Arnold (1968)
5 J Griffiths, *Dyes and Pigments*, **3**, 211 (1982)
6 B Roos, *Chem Phys Lett*, **15**, 153 (1972)
7 J-P Malrieu and D Maynau, *J Chim Phys*, **75**, 31 (1978)
8 J Griffiths, *Colour and constitution of organic molecules* London, Academic Press (1976)
9 K Nishimoto and N Mataga, *Z physik Chem* (*Frankfurt*), **12**, 335 (1957)

10 I Kanev, *J Chim Phys* , 79, 397 (1982).
11. J W Cooper, *Introduction to PASCAL for scientists* London, Wiley–Interscience (1981)
12 K. J. Bathe and E Wilson, *Numerical methods in finite element analysis* N J., Prentice-Hall (1976)
13 Y. Beppu and I Ninomiya, *Computers & Chemistry*, 6, 87 (1982).