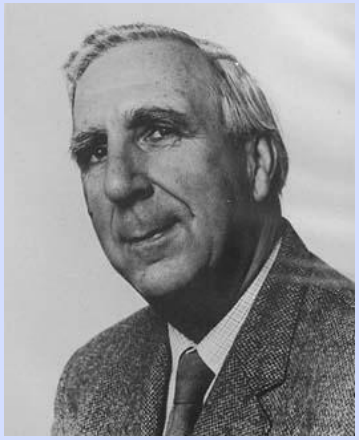
"There is more to life than increasing its speed."

Mahatma Ghandi

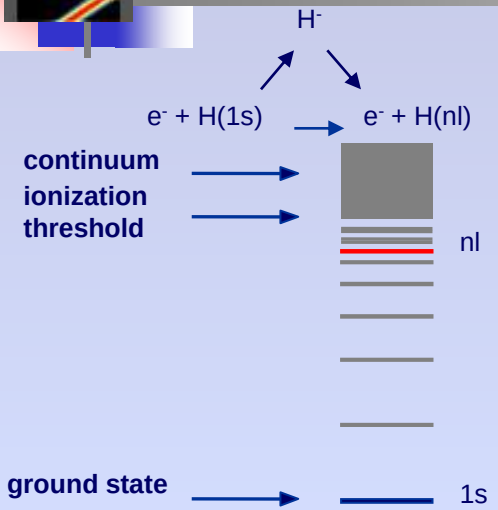
High Performance Scientific Computation is flawed

"I have little doubt that about 80% of all the results printed from the computer are in error to a much greater extent than the user would believe .."



Leslie Fox, 1971

High Performance Scientific Computation is flawed



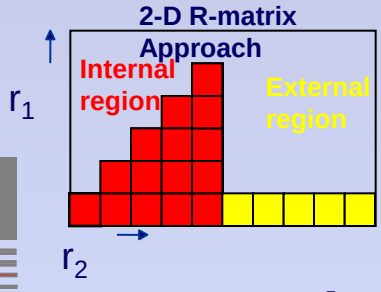
Real world

$$\Psi_E = \sum_k A_{Ek}\psi_k.$$

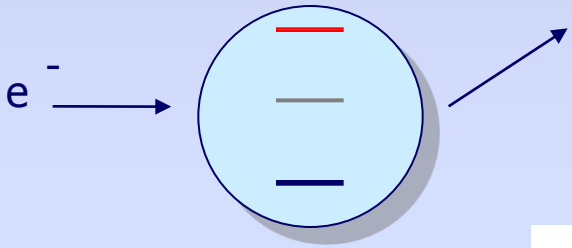
$$(\psi_i|H_{N+1}|\psi_j) = E_k^{N+1}\delta_{ij}$$

$$R_{ij} = \frac{1}{2a}\sum_k \frac{\omega_{ik}(a)\omega_{jk}(a)}{E_k^{N+1} - E}$$

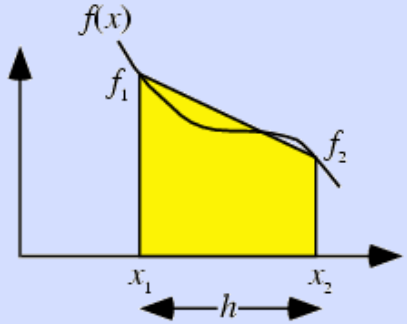
outer region



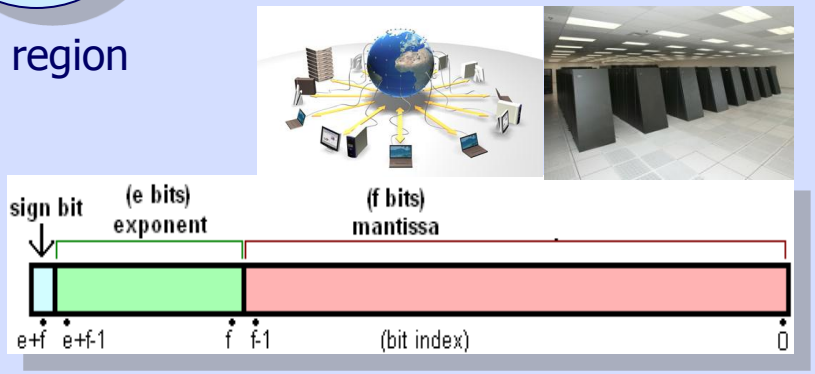
Mathematical model



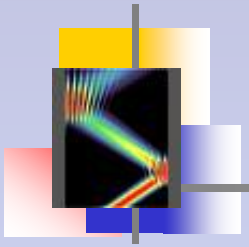
inner region



Computational model



Computer implementation

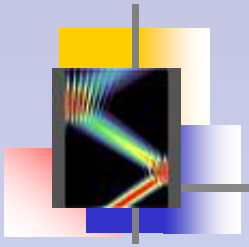


Don't rely on extended precision

$$^{\ddagger}f(x,y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

Method	f(77617,33096)
Fortran:single precision	$6.3382530 * 10^{29}$
Fortran:double precision	1.17260394005318
Fortran:quad precision	1.172603940053178631858834904520 18
VP Interval Arithmetic	[-0.827396059946821368141165095479816292005, -0.827396059946821368141165095479816291986]

‡ S.M. Rump, "Reliability in Computing. The role of Interval Methods in Scientific Computing", Academic Press, 1988.



Numerical 'health check' recommended

Ideally, we seek a numerical screening tool that will:

- report gradual and catastrophic loss of precision;
- report the accuracy of intermediate and final results;
- be of acceptable efficiency; and
- be non invasive to the source code.

Accuracy vs Precision

3.1428571 has eight decimal digit precision, irrespective of what it represents.

22/7 - accurate to eight decimal digits,

π - accurate to three decimal digits only.

The CADNA Library: a solution?

```
PROGRAM f
use cadna
double precision  :: y,x,res      type(double_st)  :: y,x,res
call cadna_init(-1)
x=77617d0  y=33096d0
res=333.75*y*y*y*y*y*y*y+x*x*(11*x*x*y*y-y*y*y*y*y*y*y- &
    121*y*y*y*y-2.0)+5.5*y*y*y*y*y*y*y*y*y+x/(2*y)
print *, res       print *, str(res)
 call cand_end()
END PROGRAM f
```

Control of Accuracy and Debugging for Numerical Applications (CADNA)

http://www.lip6.fr/cadna

The CADNA Library: numerical noise exposed

$$f(x,y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

Method	f(77617,33096)
Fortran:single precision	$6.3382530 * 10^{29}$
Fortran:double precision	1.17260394005318
Fortran:quad precision	1.17260394005317863185883490452018
VP Interval Arithmetic	[-0.82739605994682136814116509547981629 2005, -0.82739605994682136814116509547981629 1986]
CADNA: single precision	@.0 (Numerical noise)
CADNA: double precision	@.0 (Numerical noise)

How does CADNA work?

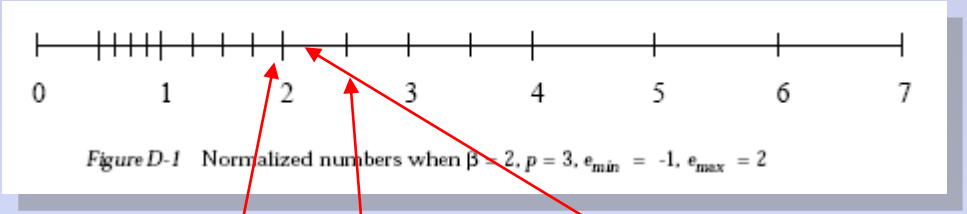


Figure D-1 Normalized numbers when $\beta = 2$, $p = 3$, $e_{min} = -1$, $e_{max} = 2$

Where no overflow occurs, the exact result, r, of any non exact floating-point arithmetic operation is bounded by two consecutive floating-point values R^- and R^+.

CADNA Library is based on the Contröle et Estimation Stochastique des Arrondis de Calculs (CESTAC) method and aims to estimate precisely the computing error in computer generated results i.e. to estimate the number of common significant figures between the computed result and the exact result.

How does CADNA work?

The CESTAC methodology

The basic idea of the method is to perform each arithmetic operation N times, randomly rounding each time, with a probability of 0.5, to R^- or R^+.

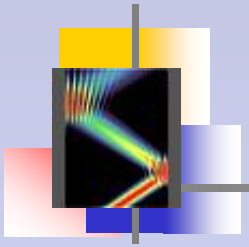

R_1=x*y


R_2=x*y


R_3=x*y

$$\bar{R} = \frac{1}{N}\sum_{i=1}^{N} R_i$$ the computed result

$$C_{\bar{R}} = f(\bar{R}, \sigma)$$ the number of exact significant digits in $\bar{R}$



How does CADNA work?

The CADNA library is an implementation of Discrete Stochastic Arithmetic (DSA) devoted to programs written in ADA, C, C++ and Fortran.

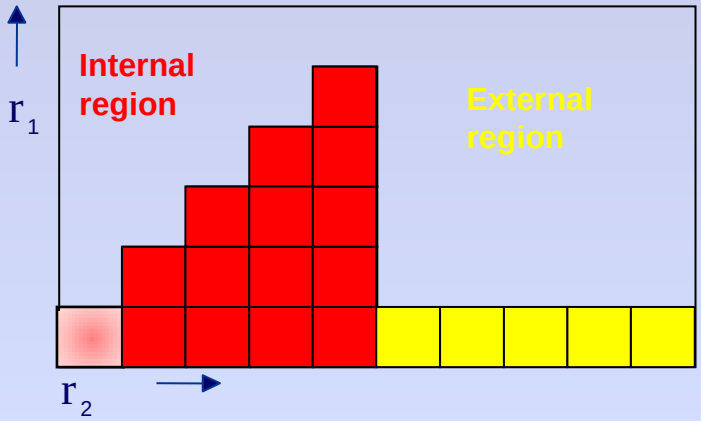
Fortran types are simply replaced by the corresponding stochastic types. The stochastic numbers are triplets containing the perturbed floating-point values.

Arithmetic operators, logical operators, all the standard intrinsic functions, as defined by the F77standard, and some vector operations have been overloaded so that when an operator is used the operands are triplets and the returned result is a triplet.

During execution when a numerical anomaly is detected, dedicated counters specific to CADNA are incremented and printed at the end of the run. These messages need to be analyzed, the source of the anomaly identified and, if necessary, the code changed.

CANDA – a diagnostic tool for legacy codes?

2-D R-matrix Approach



$$\frac{dP_{nl}}{dx^2} = \left(\frac{l(l+1)}{x^2} - \frac{2Z}{x} - k_{nl}^2 \right) P_{nl}(x), \quad x \in [a,b],$$

$$I_\lambda = J_{1,\lambda} + J_{2,\lambda},$$

Slater integrals

$$(\psi_i | H_{N+1} | \psi_j) = E_k^{N+1} \delta_{ij}$$

$$J_{1,\lambda} = \int_a^b \int_a^y f_\lambda(x,y) dx dy,$$

$$f_\lambda(x,y) = \frac{P_{n_1 l_1}(y) P_{n_3 l_3}(y)}{y^{\lambda+1}} x^\lambda P_{n_2 l_2}(x) P_{n_4 l_4}(x), \quad x \in [a,y],$$

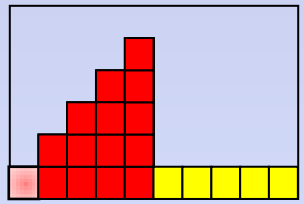
$$J_{2,\lambda} = \int_a^b \int_y^b \phi_\lambda(x,y) dx dy,$$

$$\phi_\lambda(x,y) = P_{n_1 l_1}(y) P_{n_3 l_3}(y) y^\lambda \frac{P_{n_2 l_2}(x) P_{n_4 l_4}(x)}{x^{\lambda+1}}, \quad x \in [y,b],$$

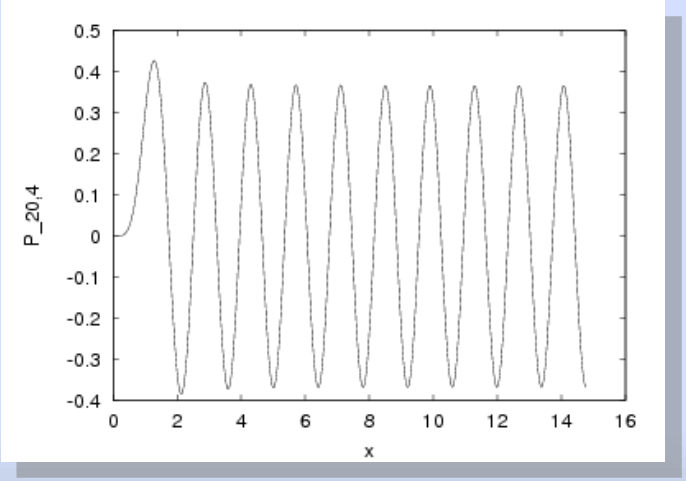
$$|l_2 - l_4| \le \lambda \le l_2 + l_4.$$

CANDA – a diagnostic tool for legacy codes?

To compute the Slater integrals 2DRMP uses, rs, a legacy subroutine that has been used R-matrix codes for over 30 years.



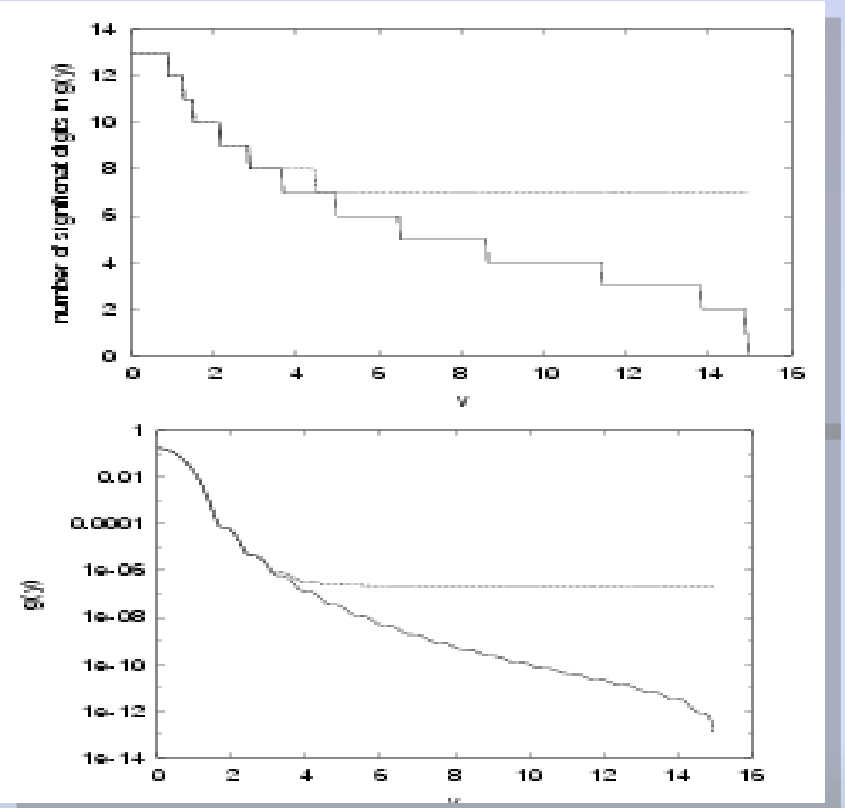
The computation of I_λ with λ in {0, 2, 4, 6, 8} in double precision for the case $a=10^{-5}$, $b=15.0$, $n_1=20$, $l_1=4$, $n_2=20$, $l_2=4$, $n_3=20$, $l_3=4$, $n_4=20$, $l_4=4$ with CADNA and using 1025 equally spaced integration points.



λ	I_λ
0	0.1247937243912E+000
2	0.471551365578E-001
4	0.288813766E-001
6	0.952431E-002
8	0.3995087E+002

CANDA – a diagnostic tool for legacy codes?

$$g(y) = \int_y^{15.0} \frac{P_{n_2,l_2}(x) P_{n_4,l_4}(x)}{x^9} dx,$$



i	$y_i = a + (i-1)h$	$g(y)$, using eqs. (14-16)
380	5.55176411132801	$0.2179550E-006$
381	5.56641253906239	$0.103235E-007$
382	5.58106096679676	$0.2177518E-006$
383	5.59570939453114	$0.100807E-007$
...	...	...
746	10.9130886621089	$0.2075749E-006$
747	10.9277370898432	$0.3474E-010$
748	10.9423855175776	$0.2075739E-006$
749	10.9570339453120	$0.3391E-010$
...	...	...
1018	14.8974610058584	$0.2075400E-006$
1019	14.9121094335928	$0.30E-012$
1020	14.9267578613272	$0.2075399E-006$
1021	14.9414062890616	$0.2E-012$

CANDA – a diagnostic tool for legacy codes?

$$g(y) = \int_y^{15.0} \frac{P_{n_2,l_2}(x) P_{n_4,l_4}(x)}{x^9} dx,$$

i	$y_i = a + (i-1)h$	$g(y)$, using eqs. (14-16)	$g(y)$, using eqs. (17-19)
380	5.55176411132801	$0.2179550E-006$	$0.104153303371283E-007$
381	5.56641253906239	$0.103235E-007$	$0.103234799220457E-007$
382	5.58106096679676	$0.2177518E-006$	$0.102120637691204E-007$
383	5.59570939453114	$0.100807E-007$	$0.100807385691787E-007$
...	...	...	...
746	10.9130886621089	$0.2075749E-006$	$0.35254005852359E-010$
747	10.9277370898432	$0.3474E-010$	$0.34742269862922E-010$
748	10.9423855175776	$0.2075739E-006$	$0.34293770930925E-010$
749	10.9570339453120	$0.3391E-010$	$0.33907324325529E-010$
...	...	...	...
1018	14.8974610058584	$0.2075400E-006$	$0.345211114012527E-012$
1019	14.9121094335928	$0.30E-012$	$0.300127939842871E-012$
1020	14.9267578613272	$0.2075399E-006$	$0.252902341589428E-012$
1021	14.9414062890616	$0.2E-012$	$0.204059815759057E-012$

Accuracy vs Precision

Even points – poor algorithm, well computed

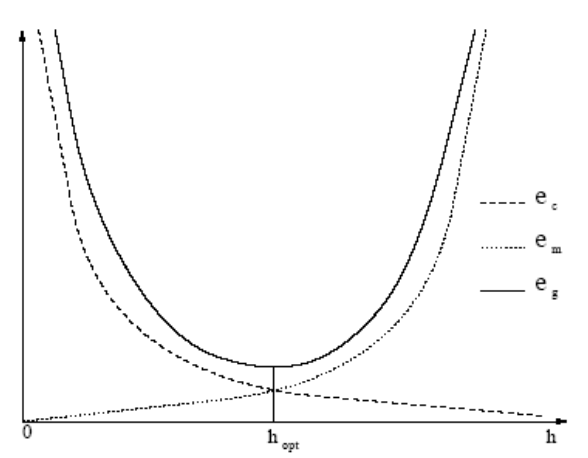
Odd points – better algorithm, poorly computed

CANDA – a diagnostic tool for legacy codes?

λ	I_λ	I_λ improved
0	0.1247937243912E+000	0.**124793**6614595E+000
2	0.471551365578E-001	0.**471551**1531988E-001
4	0.288813766E-001	0.**288813**6383162E-001
6	0.952431E-002	0.**209343**0472201E-001
8	0.3995087E+002	0.**164875**4288096E-001

Improved precision but are they accurate?

CANDA: a tool for generating benchmark results?



An approximate numerical method has a global error, $e(h)_g$, made up of a truncation error, $e(h)_m$, and a computing error, $e(h)_c$.

As h increases, $e(h)_m$ increases and $e(h)_c$ decreases.

As h decreases, $e(h)_m$ decreases and $e(h)_c$ increases.

Ideally we wish to minimize the global error. This is impossible with normal FP arithmetic as we have no estimate of $e(h)_c$. With CADNA it is possible.

CANDA: a tool for generating benchmark results?

We generate a sequence of iterations, halving the step size each time, until $|I_n - I_{n+1}| = @.0$. No further meaningful computation can occur. **The optimum result has been achieved for the method using the finite precision available.**

For Newton-Cotes type approximations it can be shown that in a series of successive iterations, if $|I_n - I_{n+1}| = @.0$, then the significant digits in common to I_n and I_{n+1} are also common to the **exact result, I, up to one bit**.

This can be used to generate benchmark results.

λ	I_λ using 2^{17} integration points
0	0.**12479372449**E+000
2	0.**47155137140**E-001
4	0.**28881377469**E-001
6	0.**20934314687**E-001
8	0.**16487550218**E-001

High performance computation of the Slater integrals

Stage 1 - method to solve the Schrödinger equation, to an accuracy of about 12 figures, in a few tens of steps using a fixed step size which is independent of n.

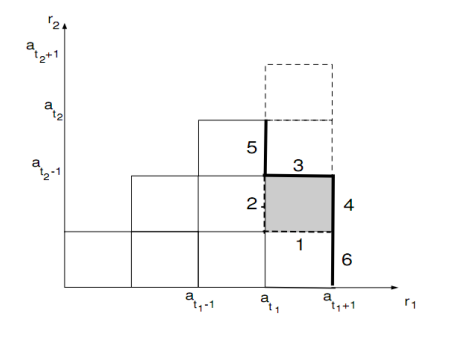
$$\frac{dP_{nl}}{dx^2} = \left(\frac{l(l+1)}{x^2} - \frac{2Z}{x} - k_{nl}^2 \right) P_{nl}(x), \quad x \in [a,b],$$

Stage 2 - using this information and by exploiting the independence of the frequencies on l, we were able to construct extended frequency dependent quadrature rules for the Slater integrals using the same mesh points.

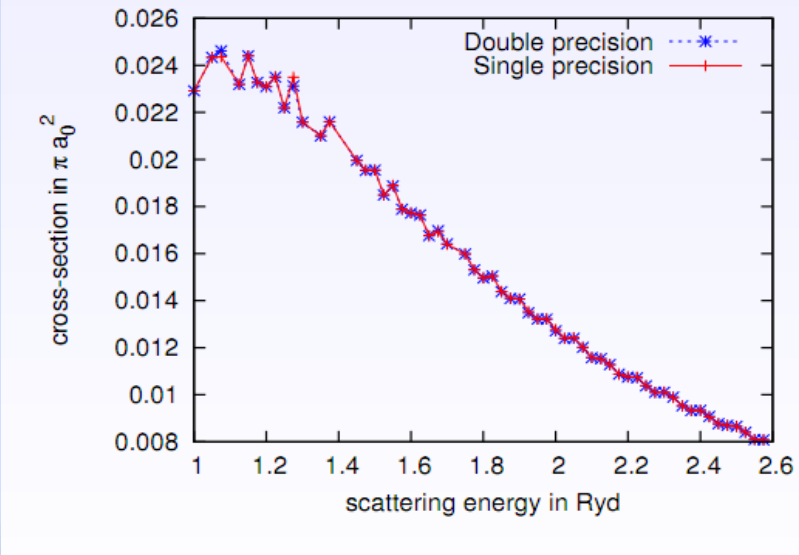
$$I_\lambda = J_{1,\lambda} + J_{2,\lambda},$$

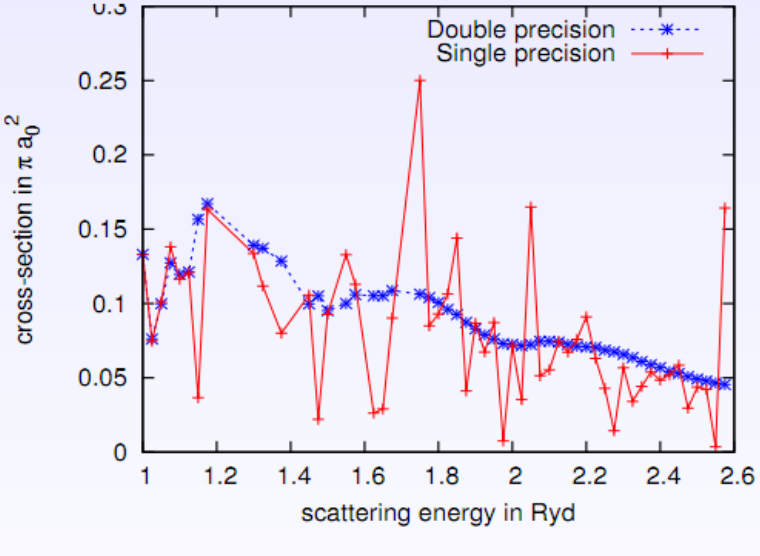
Case	# of Slater integrals	Matrix size	Newton-Cotes (secs)	EFDQR (secs)	Speedup
1	8,187,600	1680	943	38	25
2	85,741,800	5090	9880	163	61
3	302,869,500	8900	34899	472	74

CADNA: a tool to help generate energy aware algorithms?



$$
\begin{aligned}
\Re^O_{OO} &= \mathbf{r}_{OO} - \mathbf{r}_{OI}(r_{II} + \Re^I_{II})^{-1}\mathbf{r}_{IO}, \\
\Re^O_{OX} &= \mathbf{r}_{OI}(\mathbf{r}_{II} + \Re^I_{II})^{-1}\Re^I_{IX}, \\
\Re^O_{XO} &= \Re^I_{XI}(\mathbf{r}_{II} + \Re^I_{II})^{-1}\mathbf{r}_{IO}, \\
\Re^O_{XX} &= \Re^I_{XX} - \Re^I_{XI}(\mathbf{r}_{II} + \Re^I_{II})^{-1}\Re^I_{IX}.
\end{aligned}
$$





Acknowledgements

F. Jèzèquel **P. Fortin** **J.-M. Chesneaux** **J.-L. Lamotte** **R. Habel** **C. Denis**	**Laboratoire d'Informatique de Paris 6, Universitè Pierre et Marie Curie - Paris 6**
M.P. Scott	School of Mathematics and Physics, Queen's University of Belfast
L. Gr. Ixaru	Institute of Physics and Nuclear Engineering, Bucharest

"There is more to life than increasing its speed."

Mahatma Ghandi