Theor Chem Acc (2001) 105:393–399
DOI 10.1007/s002140000224

Theoretical
Chemistry Accounts

Regular article

A fast algorithm to compute atomic charges based on the topology of the electron density

P. L. A. Popelier

Department of Chemistry, UMIST, Manchester M60 1QD, UK
e-mail: pla@umist.ac.uk

Received: 1 June 2000 / Accepted: 4 October 2000 / Published online: 23 January 2001

Abstract. This work proposes a novel algorithm to compute atomic charges as defined by the theory of "atoms in molecules" (AIM). Using the divergence theorem it is possible to express the 3D volume integral over an atomic basin purely in terms of 2D surface integrals. Hence, it can be proven that an atomic charge is equal to the flux of the electric field of the whole molecule through the atom's complete boundary. This boundary consists of the interatomic surfaces and the so-called outeratomic surface, which is the open side of the atom. When fine-tuned the algorithm can generate atomic charges in the order of minutes without introducing any approximations. Moreover, the problem of the geometrical cusp occurring in atomic basins and that of multiple intersections is also eliminated. The computational overhead of computing the electric field (which is analytical) is compensated by the gain in computing time by eliminating one dimension of quadrature. The proposed algorithm opens an avenue to invalidate the oft-quoted drawback that AIM charges are computationally expensive. We explain the details of the implementation in MORPHY01 and illustrate the novel algorithm with a few examples.

Key words: Atomic change – Atoms in molecules – Numerical integration – Surface integrals

1 Introduction

The topological analysis of the electron density has proven to be an attractive method to extract chemical insight from modern ab initio wave functions. Also known as the theory of "atoms in molecules" (AIM) [1, 2], this method is currently used as an interpretative tool in many areas ranging from high-resolution crystallography to mineralogy, transition-metal chemistry and biochemistry [3]. Although software to obtain atomic properties has been available since the 1980s [4, 5] most AIM applications incorporate only critical point information and visual representations of the electron density gradient vector field. Of the atomic properties, such as energy, dipolar polarisation and distributed polarisability, the atomic charge has received most attention. This is understandable since atomic charges continue to play a dominant role in the construction of force fields and in the chemical interpretation of charge distribution and charge transfer inside a molecule (including van der Waals complexes). Over the years many population methods have been designed to meet a need for purpose-designed atomic charges. Occasionally the AIM theory is mistaken to be a population method although it has never been developed with that goal in mind. Instead, since its early development as a virial partitioning scheme it has its roots in quantum mechanics and generates atomic charges almost as a by-product.

In this article we address an issue that has plagued AIM since its inception; namely, the computational cost, accuracy and robustness of atomic integration algorithms. Compared to other methods to obtain atomic charges, such as the Mulliken or the natural population analysis [6], one needs sophisticated algorithms to obtain AIM charges in a reasonable time. Since the first algorithms [4, 5] for atomic integration became available progress [7–10] has been made in terms of robustness, accuracy and speed. The resulting algorithms were implemented in programs such as GAUSSIAN98 [11], MORPHY98 [12] and AIMPAC [13]. They have in common that they obtain the atomic charge by a volume integral of the electron density over the so-called atomic basin, the region of 3D space containing the gradient paths attracted by the nucleus inside the atom. The current algorithm obtains the atomic charge via surface integrals only.

Contribution to the Proceedings of the 2000 Symposium on Chemical Bonding: State of the Art in Conceptual Quantum Chemistry

In the next section we explain the essential idea behind the new algorithm and its detailed implementation is explained in Sect. 3, using water as a pilot system. Subsequently, in Sect. 4, we illustrate the advantage of the current algorithm for atoms with geometrical cusps, such as carbon in methanal.

2 Atomic charges from surface integrals

In this section we explain the main idea behind the new integration algorithm. The divergence theorem [14] states that the flux of a vector field **A** out of a closed surface $\partial\Omega$ equals the 3D integral of the divergence of **A** over the volume enclosed by the surface $\partial\Omega$ or

$$\iiint_{\Omega} \mathrm{d}V(\nabla \bullet \mathbf{A}) = \iint_{\partial\Omega} \mathrm{d}S(\mathbf{A} \bullet \mathbf{n}) \;, \tag{1}$$

where $\partial\Omega$ is the oriented closed surface that bounds Ω. The proof of this theorem essentially reduces to a cancellation of one of the integrals in the triple (volume) integral with the differential operator of the divergence. As a result we are left with only two integrals constituting the surface integral on the right-hand side of Eq. (1). This means that it is possible to eliminate an entire dimension of integration (2D instead of 3D) provided one can write the integrand of the 3D integral as the divergence of a vector field. This general problem is formulated as follows.

$$\text{Given} \iiint \mathrm{d}V f(\mathbf{r}),\ \text{determine } \mathbf{A} \text{ such that } \nabla \bullet A = f(\mathbf{r}). \tag{2}$$

If $f(\mathbf{r})$ equals a continuous charge distribution, denoted by $\rho_{\mathrm{charge}}(\mathbf{r})$, the differential form of Gauss's law of electrostatics provides the solution to the problem stated in Eq. (2), since

$$\nabla \bullet (\mathbf{E}/4\pi) = \rho_{\mathrm{charge}}(\mathbf{r}), \tag{3}$$

where **E** is the electric field generated by the charge distribution. In our case $\rho_{\mathrm{charge}}(\mathbf{r})$ is the total molecular charge distribution involving both the electronic and the nuclear contributions:

$$\rho_{\mathrm{charge}}(\mathbf{r}) = (-e)\rho(\mathbf{r}) + e\sum_{A} Z_A \delta(\mathbf{r} - \mathbf{R}_A), \tag{4}$$

where $\rho(\mathbf{r})$ is the electron density, e is the elementary charge and Z_A is the atomic number of nucleus A. The volume integral of $\rho_{\mathrm{charge}}(\mathbf{r})$ over a given atomic basin, Ω, yields the atomic charge, $q(\Omega)$, since from Eq. (4) we obtain

$$\iiint_{\Omega} \mathrm{d}V\, \rho_{\mathrm{charge}}(\mathbf{r}) = (-e)\iiint_{\Omega} \mathrm{d}V\rho(\mathbf{r}) + e\sum_{A} Z_A \iiint_{\Omega} \mathrm{d}V\delta(\mathbf{r} - \mathbf{R}_A) = (-e)N(\Omega) + eZ_{\Omega} = q(\Omega), \tag{5}$$

where $N(\Omega) = \int_{\Omega} \mathrm{d}V\, \rho(\mathbf{r})$ is the atomic population and the Dirac delta function, δ, picks out the nuclear charge inside a given atomic basin, Ω. From here on we work in atomic units, which means that e is replaced by 1.

If the vector field, **A**, is set equal to the total electrostatic field, **E**, of the whole molecule (divided by 4π) we can use the divergence theorem (Eq. 1) to write the atomic charge, $\mathbf{q}(\Omega)$, as

$$q(\Omega) = \iiint_{\Omega} \mathrm{d}V\rho_{\mathrm{charge}}(\mathbf{r}) = \frac{1}{4\pi}\iiint_{\Omega} \mathrm{d}V(\nabla \bullet \mathbf{E}_{\mathrm{tot}}) = \frac{1}{4\pi}\iint_{\partial\Omega} \mathrm{d}S(\mathbf{E}_{\mathrm{tot}} \bullet \mathbf{n}), \tag{6}$$

where $\mathbf{E}_{\mathrm{tot}} = \mathbf{E}_{\mathrm{elec}} + \mathbf{E}_{\mathrm{nuc}}$ is the total molecular electric field due to the electronic and nuclear charge distribution. The field $\mathbf{E}_{\mathrm{elec}}$ is computed [15] via

$$\mathbf{E}_{\mathrm{elec}}(\mathbf{r}) = -\nabla_{\mathbf{r}} V_{\mathrm{elec}}(\mathbf{r}) = -\nabla_{\mathbf{r}} \iiint_{\text{all space}} \frac{-\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}\mathrm{d}\mathbf{r}' = -\iiint_{\text{all space}} \rho(\mathbf{r}')\frac{(\mathbf{r} - \mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|^3}\mathrm{d}\mathbf{r}', \tag{7}$$

and the field, $\mathbf{E}_{\mathrm{nuc}}$, is simply

$$\mathbf{E}_{\mathrm{nuc}}(\mathbf{r}) = \sum_{A} \frac{Z_A(\mathbf{r} - \mathbf{r}_A)}{|\mathbf{r} - \mathbf{r}_A|^3}, \tag{8}$$

where $\mathbf{r}_A$ is the position vector of the nucleus A.

It is important to realise that the molecular electric field can be obtained from analytical integrals. The electron density of a free molecule occupies all space and hence integrals occurring in derived properties such as the electric field have infinite boundaries, which are much easier to tackle than finite boundaries. Moreover, recent advances in integral evaluation [16] justifies the computational overhead generated by the electric field compared to that of the electron density. In view of the finite boundaries of the integrals a completely analytical integration of an atomic property is prohibitive [7].

Note that surface integration can, in principle, also be used to compute an atomic volume. If we set $f(\mathbf{r})$ equal to 1 in Eq. (2) then **A** becomes $1/3\mathbf{r}$ because $\nabla\bullet\mathbf{r} = 3$. Hence the flux of $1/3\mathbf{r}$ through the atom's boundary, $\partial\Omega$, equals its volume. This method is computationally very attractive since an atom's volume is very sensitive to the quality of a 3D integration, especially when cusps and long tails occur in the atom's shape [17]. It should be possible to solve Eq. (2) for other atomic properties of interest, such as the kinetic energy or the dipole moment. If the unknown vector field, **A**, is written as the gradient of a scalar function, *B*, then the divergence equation $\nabla\bullet\mathbf{A} = f(\mathbf{r})$ reduces to the Poisson equation $\nabla\bullet\nabla B = \nabla^2 B = f(r)$, which can be solved using Green's functions. Bader and Gatti have discussed [18] the use of Green's functions before in the context of AIM but for a different purpose.

We now discuss in detail how surface integration can be used to obtain atomic charges for a given atomic basin, Ω.

3 The algorithm

3.1 The boundary of the atom

In general, the boundary, ∂Ω, of an atom in a free molecule consists of the union of its interatomic surfaces (IASs). An IAS or zero-flux surface consists of a bundle of gradient paths originating at infinity and attracted by a bond critical point (BCP). A BCP is a type of critical point (a point where $\nabla\rho = \mathbf{0}$), roughly between two nuclei, such that the electron density is a local maximum in the IAS and a local minimum with respect to the direction perpendicular to the IAS. For a free molecule the IASs extend to infinity, so in principle we could represent ∂Ω by the IASs only, and regard infinity as the boundary of the open side of the atom. However, gradient paths cannot be traced accurately in the far outer regions of a molecule (i.e. path lengths in excess of 15 au) because Gaussian-expanded wave functions do not show the correct asymptotic behaviour. Moreover, the electric field does not decay rapidly to zero in the outer regions of the molecule. As a result we bound the atom at its open side by an envelope, which we call the outeratomic surface (OAS). The total atomic boundary, ∂Ω, is then the union of the OAS and all IASs bounding the atom inside the molecule, or

$$\partial\Omega = \{\mathrm{IAS_i}\} \cup \mathrm{OAS}, \qquad (9)$$

where $\{\mathrm{IAS_i}\}$ denotes the set of all IASs, each $\mathrm{IAS_i}$ being centred at the corresponding *i*th BCP. In order to obtain the atomic charge one needs to compute the flux of the total molecular electric field through ∂Ω.

The surfaces involved in the integration of oxygen in water are shown in Fig. 1. The oxygen atom is bounded by two IASs, each forming the boundary between the oxygen and one hydrogen atom. The large open side of oxygen constitutes the OAS, which is represented by a part of a sphere centred on the oxygen nucleus. This sphere is called the integration sphere (IS) and it approximates an envelope of constant electron density typically with a value of $\rho = 10^{-7}$–10^{-4} au. In the example shown in Fig. 1 the electron density cutoff value is chosen to be 10^{-7} and the concomitant radius of the IS is 5.4 au. Clearly the radius depends on the atomic number of the nucleus and on the chosen contour value. The fact that the OAS is represented by a simple object such as a sphere is not crucial. More sophisticated representations, conveniently given by $r = f(\theta, \varphi)$ are necessary if one wishes to compute an atomic volume for example, because a typical atomic volume requires the knowledge of the $\rho = 0.001$ au envelope. The only important issue in the context of the atomic charge is that the OAS enables the encompassment of virtually the entire electron density contained in the atomic basin. In other words, if the radius of the IS is too small, some charge will not be included in the basin. The task of the IS is just to provide a simple and adequate boundary of the atom at its open end, i.e. on the OAS.

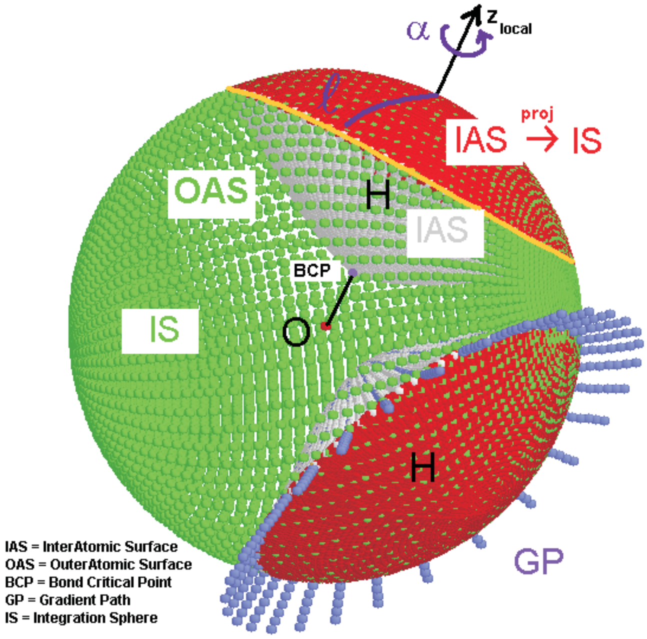

Fig. 1. The atomic charge of oxygen in water via surface integrals. The oxygen atom is bounded by two interatomic surfaces (*IASs*, *grey*) and the outeratomic Surface (*OAS*, *pure green*). The integration sphere (*IS*, *green* and *green/red*) has a radius of 5.4 au determined by a $\rho = 10^{-7}$ au cutoff. Each IAS is projected onto the IS (*red*). The flux through this part of the IS is subtracted from the flux through the whole IS. The boundary of one IAS is shown in *amber*. It consists of the intersection of the IAS with the IS, and is also the boundary of the IS-projected IAS patch (*red*). The *z*-axis of the local axes system describing this patch is shown, together with its independent surface parameters α (angle) and ℓ (path length on IS)

The OAS can be thought of as a sphere punctured by the IASs. As explained in the next section, it is computationally attractive to introduce the projection of an IAS onto the IS. This projection is shown as a red surface patch in Fig. 1. The OAS is simply the remainder of the IS after the projections of each IAS onto the IS have been subtracted, or formally

$$\mathrm{OAS} = \mathrm{IS} \backslash \{\mathrm{IAS_i} \rightarrow \mathrm{IS}\}, \qquad (10)$$

where → IS denotes the projection of an IAS onto the IS. Owing to the representation of the OAS as a difference set, the flux of the electric field through the IS-projected IASs is also literally subtracted from the flux through the IS.

3.2 Surface representation

It is advantageous to represent a surface of arbitrary shape in terms of two independent parameters u and v. A parameterised surface is a mapping from (u, v) space to 3D space (x_1, x_2, x_3) or more precisely, a function $\Phi: D \subset \mathbb{R}^2 \to \mathbb{R}^3$, where D is some domain in $\mathbb{R}^2$. The surface, S, corresponding to the function Φ is its image, i.e. $S = \Phi(D)$ and $\Phi(u, v) = [x_1(u, v), x_2(u, v), x_3(u, v)]$. In previous work [19] the IAS has been represented in this manner.

An IAS is constructed by tracing gradient paths in the reverse direction, away from a small circle of initial points centered at the BCP and lying in a plane locally tangent to the IAS. The independent parameter u is the angle, α, fixing the initial points on the circle and v is ℓ, which is the path length of a gradient path on the IAS. An analytical representation of an IAS is obtained by fitting Chebyshev polynomials $T_n(\ell)$ to the IAS gradient paths, which have been computed by the numerical Runge–Kutta–Cash–Karp method [20]. The expansion coefficients, which depend on the angular parameter α are in turn expanded as a Fourier series. So the function $\Phi_{\mathrm{IAS}}(\alpha, \ell)$ is given by

$$x_i(\alpha,\ell)=\sum_m\sum_n\left[\mathrm{d}_{inm}\cos(m\alpha)+\mathrm{d}'_{inm}\sin(m\alpha)\right]T_n(\ell). \quad (11)$$

The OAS is actually a punctured sphere (Fig. 1) represented as a domain $D \subset [0, \pi] \times [0, 2\pi]$ determined by the 3D shape of the IASs. The amber line in Fig. 1 marks the boundary of one IAS, and hence constitutes the edge of the "hole" that this IAS makes in the IS. The edge of the hole in the IS appears as an arbitrary curve in (θ, φ) space, possibly displaying cusps. One can show that installing a 2D quadrature grid directly on the OAS is very hard and is riddled with numerical pitfalls. We explain later that it is better to integrate over the "holes" first and then subtract it from the integration over full (θ, φ) space, i.e. the IS.

If we represent the IS (i.e. the full sphere of which the OAS is a subset) simply via the definition of polar spherical coordinates, then $\Phi_{\mathrm{IS}}(\theta, \varphi)$ is given by

$$\begin{aligned} x_1(\theta,\varphi) &= r_{\mathrm{IS}}\sin\theta\cos\varphi,\\ x_2(\theta,\varphi) &= r_{\mathrm{IS}}\sin\theta\sin\varphi,\\ x_3(\theta,\varphi) &= r_{\mathrm{IS}}\cos\theta, \end{aligned} \quad (12)$$

where r_{IS} is the radius of the IS.

The projection of an IAS onto the IS is determined by a mapping from a local coordinate system (α, ℓ) to (x_1, x_2, x_3), where – in analogy with the IAS coordinate system – ℓ is a radial parameter and α is an angular parameter. Figure 1 (generated using MORPHY01 [21] and RasMol [22]) illustrates the details of this coordinate system. The local z-axis goes through the oxygen nucleus and the BCP corresponding to the IAS, oriented away from the nucleus. The boundary of the IAS (amber curve in Fig. 1) is where the IAS intersects the IS. This boundary is also the boundary of the IAS projected onto the IS (red patch). The local z-axis intersects this patch at a point which can be taken as the natural centre of the patch (analogous to the BCP for the IAS). From this point, ℓ measures the length of a path (purple) on the IS terminating in an arbitrary point on the red patch. The angular parameter α enables this path to reach any point on the red patch. It is clear that for each IS-projected IAS we have a mapping, given by $\Phi_{\mathrm{IAS}\to\mathrm{IS}}(\alpha, \ell)$ described by the following equations:

$$\begin{aligned} x_1(\alpha,\ell) &= r_{\mathrm{IS}}\sin(\ell/r_{\mathrm{IS}})\cos\alpha,\\ x_2(\alpha,\ell) &= r_{\mathrm{IS}}\sin(\ell/r_{\mathrm{IS}})\sin\alpha,\\ x_3(\alpha,\ell) &= r_{\mathrm{IS}}\cos(\ell/r_{\mathrm{IS}}). \end{aligned} \quad (13)$$

3.3 Surface integration

The flux of a vector field $\mathbf{A}$ through a surface S is computed in (u, v) space over a domain D that is determined by the boundary of S in real space:

$$\mathrm{Flux} = \iint\limits_S \mathrm{d}S(\mathbf{A}\bullet\mathbf{n}) = \iint\limits_D \mathrm{d}u\mathrm{d}v\ \mathbf{A}\bullet(\mathbf{T}_u\times\mathbf{T}_v), \quad (14)$$

where $\mathbf{T}_u \times \mathbf{T}_v$ is the normal to the oriented surface S, and the tangent vectors $\mathbf{T}$ are given by $\mathbf{T}_u = \sum_i \frac{\partial x_i(u,v)}{\partial u}\mathbf{u}_i$ and $\mathbf{T}_v = \sum_i \frac{\partial x_i(u,v)}{\partial u}\mathbf{v}_i$ where $\mathbf{u}_\mathrm{i}$ is the x, y, or z unit vector. Note that the normal has to point outwards, away from the nucleus. This is ensured by checking if the dot product between the normal at the BCP and a vector from the nucleus to the BCP is positive. If not, the sign of the flux has to be reversed. The flux through the total boundary of the atom, $\partial\Omega$, consists of three types of contributions coming from the IS, the set of IASs and the projection of each IAS onto the IS. The computation of the normal $\mathbf{T}_u \times \mathbf{T}_v$ is analytical and straightforward since each of the three types of boundary surfaces is described by an analytical expression.

In practice Eq. (14) is evaluated using a 2D Gauss–Legendre quadrature. This technique has been applied before in a differential geometrical study of the IAS [23].

3.4 Implementation

A flow chart of the most important subroutines of the atomic charge algorithm as it is implemented in MORPHY01 [21] is shown in Fig. 2. A concise description of each subroutine's task is given in Table 1. Ignoring numerous (important) details the algorithm (controlled by the master subroutine SINTEGRAL) basically works as follows.

The user determines a cutoff value for ρ, which is the practical edge of the atom at its open end. From this cutoff value the radius of the IS is determined and $\Phi_{\mathrm{IAS}\to\mathrm{IS}}(\alpha, \ell)$ is found for each IAS (ANALOAS). The radius of the IS is first set to the distance between the nucleus in the integrated basin and the nearest point on the constant ρ envelope. If the IS is near another nucleus, large electric fields arise, which cause numerical

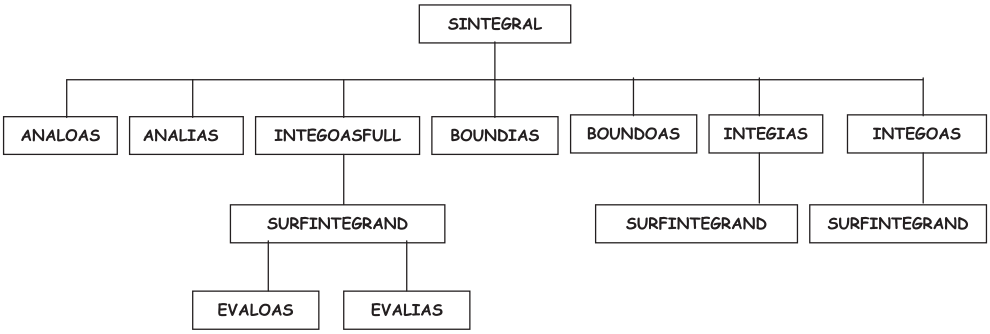

Fig. 2. Simplified flow diagram of the subroutines constituting the surface integration procedure that computes atomic charges. A subroutine is called by another one if it is one level up and connected to it. The subroutines are executed from left to right. Table 1 contains a brief description of the task that each subroutine performs. To avoid clutter in this diagram the calls to EVALOAS and EVALIAS in SURFINTEGRAND are shown only once

Table 1. Brief description of the subroutine used in the surface integration algorithm to obtain atomic charges. The interdependence of the subroutines is shown in Fig. 2

Module	Task
SINTEGRAL	Controls atomic integrations via surface integration (master subroutine)
ANALOAS	Finds the radius of the integration sphere, generates a mapping between the Cartesian coordinates with respect to the global frame and the local (α, ℓ) coordinate system associated with the projection of each IAS patch onto the integration sphere
ANALIAS	Generates a Fourier–Chebyshev expansion (analytical expression) for a given interatomic surface
INTEGOASFULL	Performs a surface integration over the full integration sphere
BOUNDIAS	Computes the boundaries of a given interatomic surface determined by its intersection with the integration sphere and fits splines to the boundary
BOUNDOAS	Computes the boundary of an integration-sphere-projected interatomic surface patch in terms of the local coordinate system, using spline functions
INTEGIAS	Performs a surface integration over a given interatomic surface via Gauss–Legendre quadrature
INTEGOAS	Performs a surface integration over an integration-sphere-projected interatomic surface patch. This surface integral contribution will be subtracted from the surface integral over the integration sphere computed by INTEGOASFULL
SURFINTEGRAND	Calculates the integrand of the surface integral for a given quadrature point
EVALIAS	Computes the Cartesian coordinates of a point on an interatomic surface for a given (α, ℓ) value and the corresponding normal vector to the interatomic surface in that point
EVALOAS	Computes the Cartesian coordinates of a point on a given integration-sphere-projected interatomic surface patch for a given (α, ℓ) value and the corresponding normal vector in that point

inaccuracies in the integration of the IS. This is why the radius of the IS is adjusted until it avoids all external nuclei by a preset threshold distance. Then, IAS gradient paths are traced up to a ρ value somewhat smaller than the cutoff, and an analytical expression is fitted to the IAS (ANALIAS). Consequently the boundary of each IAS is determined by finding the intersection between it and the IS, which results in the amber curve in Fig. 1. This annular boundary can be expressed as a function $\ell_{\mathrm{IAS}} = f(\alpha_{\mathrm{IAS}})$, which is conveniently approximated by a spline algorithm [20] (in BOUNDIAS). Then, the boundary of the IS-projected IAS is computed in terms of its own local coordinates (α_{OAS}, ℓ_{OAS}). The boundary of the IS-projected IAS is identical to the boundary of the corresponding IAS (Fig. 1). In a similar way the boundary function $\ell_{\mathrm{IAS}\rightarrow\mathrm{IS}} = g(\alpha_{\mathrm{IAS}\rightarrow\mathrm{IS}})$ is again approximated by a spline algorithm (in BOUNDOAS) in terms of the local IS-projected coordinates.

Now that the domains of the IASs and their projections onto the IS have been determined, the surface integration can be performed (using INTEGIAS and INTEGOAS). The integration over the IS can be performed independently from knowledge of the IAS boundaries since it occurs over the whole integration sphere.

Note that the surface-based integration algorithm requires a proper analytical expression for surfaces that contain ring critical points or “gaps” [8]. Preliminary work in progress has shown that such expression can, in principle, be obtained except for IASs appearing in an unstable conflict or bifurcation structure.

4 Examples

A simple system that demonstrates the advantage of the current method is methanal, $H_2C = O$. The carbon atom in methanal is trigonal and possesses three long cusplike tails. It has been observed that atoms with (geometrical) cusps require large quadrature grids in 3D integration in order to reduce the integration error, $L(\Omega)$, although the size of the grid is not always a guarantee for a low $L(\Omega)$ value [17]. In a typical type of 3D integration (excluding the type using natural coordinates [4]) the atomic boundary is probed by a vector centred at the nucleus. This vector is called an integration ray and sweeps through all possible directions and may intersect an IAS once or more. Detecting cusps in the atom's geometry requires very fine probing. Moreover, finding multiple intersections between an integration ray and the atomic boundary can take an inordinate amount of computing time if the analytical expression of the IAS is not known [8]. In the current method the effect of the cusp is eliminated because each IAS is integrated regardless of its geometrical behaviour with respect to another IAS. In other words, the boundary of the atom is not screened from the nucleus – as is the case with 3D integration – but each IAS contributes its own part to the total flux without reference to another IAS. A surface integration for all atoms in methanal yields atomic charges that add up to 9×10^{-6} au. This result was obtained in just under 1 min using a Compaq/DEC 666 MHz personal workstation. On the other hand, the corresponding 3D integration yields atomic charges that add up to 3×10^{-5} au (with a very large grid on the carbon) requiring over 17 min computing time. Currently the electric field is calculated externally to MORPHY01 using the very fast PRISM algorithm [16] implemented in GAUSSIAN98 [11]. It is possible that in the future a simpler but less efficient implementation will be internally added to MORPHY01, based on the integral recurrence algorithm of Obara and Saika [24]. The ratio between the computing time for the 3D and the 2D integration is about 20, but does not reflect the true gain because the calculations are dominated by the computing time required to construct the IASs.

Another example is given by the integration of a central carbon (C_γ) in a conformation of the amino acid arginine calculated at the HF/6-31G* level, involving 392 primitives, 26 nuclei and 47 molecular orbitals. The surface integration yielded $q(\Omega) = 0.554$ in only 19.5 min. Note that the 3D integration involved just under a million quadrature points and the 2D integration only about 57,000. With the PRISM algorithm it took only 5 min to evaluate the electric field in these 57,000 quadrature points. The fact that only one quarter of the time is spent on the evaluation of the electric field warrants further research into the efficient and accurate representation of IASs and the OAS. This work is currently in progress.

Current work is directed at a better representation of an IAS, one which not only consumes less computing time but also one which does not break down when the IAS contains ring critical points or when the BCPs shows a high ellipticity [8].

Finally, it should be mentioned that the proposed method of calculating atomic charges is not only a computational improvement but also provides an avenue to reinterpret AIM atomic charges. Indeed, the molecular electric field, **E**, has zero-flux surfaces of its own, bounding basins in **E** that are neutral. The AIM atomic charge results from the gradient vector field of ρ carving out a portion of space in **E.** If the topology of $\nabla\rho$ and **E** is very similar then atomic charges are expected to be nearly vanishing, because a basin in $\nabla\rho$ will almost coincide with a basin in **E.** The potential mismatch between the topologies of **E** and $\nabla\rho$ may shed light on the nature of AIM atomic charges.

5 Conclusion

A new algorithm is proposed that calculates AIM atomic charges using surface integrals only. The divergence theorem expresses the atomic charge as the flux of the total molecular electric field through the boundary of the atomic basin. Since the molecular electric field can be calculated analytically by very fast algorithms, and since the surface integration requires 1–2 orders of magnitude fewer quadrature points, an atomic charge can now be obtained within several minutes rather than hours on a typical personal computer or workstation for reasonably sized systems. This approach also contributes to the robustness and accuracy of atomic integration because it does not suffer from the cusp problem or the multiple intersection problem. Besides a reduction in the number of quadrature points the new algorithm should provide a more straightforward route to obtain atomic charges.

Acknowledgements. W. den Otter is thanked for a useful discussion. The referee is thanked for pointing out an interesting paper [25], which refers to work [26] that mentioned expressing an atomic charge as the flux of the electric field through its interatomic boundaries.

References

1. Bader RFW (1990) Atoms in molecules: a quantum theory. Clarendon, Oxford
2. Popelier PLA (2000) Atoms in molecules. An introduction. Pearson Education, Harlow
3. Popelier PLA, Aicken FM, O'Brien SE (2000) In: Hinchliffe A (ed) Chemical modelling: applications and theory, vol 1. Royal Society of Chemistry Specialist Periodical Report. Royal Society of Chemistry, Cambridge, pp 143–198
4. Biegler-König FW, Nguyen-Dang TT, Tal Y, Bader RFW, Duke AJ (1981) J Phys B At Mol Op Phys 14: 2739
5. Biegler-König FW, Bader RFW, Tang TH (1982) J Comput Chem 3: 317
6. Reed AE, Weinstock RB, Weinhold F (1985) J Chem Phys 83: 735
7. Popelier PLA (1996) Mol Phys 87: 1169
8. Popelier PLA (1998) Comput Phys Commun 108: 180
9. Stefanov BB, Cioslowski J (1995) J Comput Chem 16: 1394
10. Cioslowski J, Nanayakkara A, Challacombe M (1993) Chem Phys Lett 203: 137
11. Frisch MJ, Trucks GW, Schlegel HB, Scuseria GE, Robb MA, Cheeseman JR, Zakrzewski VG, Montgomery JA Jr, Stratmann RE, Burant JC, Dapprich S, Millam JM, Daniels AD, Kudin KN, Strain MC, Farkas O, Tomasi J, Barone V, Cossi M, Cammi R, Mennucci B, Pomelli C, Adamo C, Clifford S, Ochterski J, Petersson GA, Ayala PY, Cui Q, Morokuma K, Malick DK, Rabuck AD, Raghavachari K, Foresman JB, Cioslowski J, Ortiz JV, Baboul AG, Stefanov BB, Liu G, Liashenko A, Piskorz P, Komaromi I, Gomperts R, Martin RL, Fox DJ, Keith T, Al-Laham MA, Peng CY, Nanayakkara A, Gonzalez C, Challacombe M, Gill PMW, Johnson B, Chen W, Wong MW, Andres JL, Gonzalez C, Head-Gordon M, Repl-

ogle ES, Pople JA (1998) Gaussian 98, revision A.7. Gaussian, Pittsburgh, Pa

12. Popelier PLA with a contribution from Bone RGA (1998) Morphy98. UMIST, Manchester, UK, http: //www.ch.umist.ac.uk/morphy
13. AIMPAC (a program suite for the topological analysis of the electron density) Bader group, McMaster University, Canada
14. Marsden JE, Tromba AT (1981) Vector calculus. Freeman, San Francisco
15. Jackson JD (1975) Classical electrodynamics. Wiley, New York
16. Johnson BG, Gill PMW, Pople JA (1993) Chem Phys Lett 206: 239
17. Aicken FM, Popelier PLA (2000) Can J Chem 78: 415
18. Bader RFW, Gatti C (1998) Chem Phys Lett 287: 233
19. Popelier PLA (1994) Theor Chim Acta 87: 465
20. Press WH, Flannery BP, Teucholsky SA, Vetterling WT (1992) Numerical recipes. Cambridge University Press, Cambridge
21. Popelier PLA, with a contribution from Bone RGA, Kosov D, Panhuis M (2001) Morphy01. UMIST, Manchester, UK
22. Sayle R (1994) RasMol (a molecular visualisation program). Glaxo, Stevenage, UK
23. Popelier PLA (1996) Can J Chem 74: 829
24. Obara S, Saika A (1986) J Chem Phys 84: 3963
25. Bader RFW, Bayles D, Heard GL (2000) J Chem Phys 112: 10095
26. Bader RFW, Keith TA (1993) J Chem Phys 99: 3683