

# libecpint: A C++ library for the efficient evaluation of integrals over effective core potentials

Robert A. Shaw<sup>1</sup> and J. Grant Hill<sup>1</sup>

<sup>1</sup> Department of Chemistry, University of Sheffield, Sheffield S3 7HF, UK

DOI: [10.21105/joss.03039](https://doi.org/10.21105/joss.03039)

## Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Jack Poulson](#) ↗

## Reviewers:

- [@felipeZ](#)
- [@lorenzo-rovigatti](#)

Submitted: 02 February 2021

Published: 15 April 2021

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

Effective core potentials (ECPs) are widely-used in computational chemistry both to reduce the computational cost of calculations, ([M. Dolg, 2000](#)) and include relevant physics that would not otherwise be present ([M. Dolg, 2002](#)). In particular, for heavy main-group atoms ([Wadt & Hay, 1985](#)) and transition metals ([Hay & Wadt, 1985](#)), the number of core electrons greatly outnumbers the number of valence electrons. It is generally considered that these will not play a significant role in chemical reactivity, and thus can be frozen. Moreover, these electrons show significant relativistic character ([M. Dolg, 2002](#); [M. Dolg & Cao, 2012](#)). Both of these issues can be resolved with the introduction of an effective core, represented as a fixed electronic potential. This potential is typically represented as a linear combination of gaussians of varying angular momenta ([M. Dolg, 2000](#)).

The introduction of an ECP results in an additional term in the core Hamiltonian, over which new electronic integrals must be computed. These three-center integrals are far from trivial, and they cannot in general be treated the same way as other electronic integrals ([Flores-Moreno et al., 2006](#); [McMurchie & Davidson, 1981](#)). Several widely used computational chemistry codes lack the ability to calculate these integrals due to the difficulty involved in their computation. The present library, `libecpint`, provides an open-source solution to this. It is a standalone library written in modern C++ capable of the highly efficient computation of integrals over ECPs with gaussian orbitals of arbitrary angular momentum, along with their first and second geometric derivatives. The methods implemented are based on novel algorithms that use automatic code generation and symbolic simplification of recursive expressions, along with highly optimised Gauss-Chebyshev quadrature.

## Statement of need

Effective core potentials are an essential part of modern computational chemistry. However, existing implementations are typically unavailable or inaccessible for free use by the open source community. Commonly used proprietary software, such as Gaussian ([Frisch et al., 2016](#)) or Molpro ([Werner et al., 2019](#)), do not make details of their implementations available, while the few open-source computational chemistry packages either do not include ECP functionality or use outdated implementations that would not be compatible with modern codebases. A notable example of this is the widely-used Psi4 package ([Smith et al., 2020](#)), in which a rudimentary version of `libecpint` was originally implemented. Prior to this, the inclusion of ECPs was one of the most requested features by the user base.

Additionally, there has been a recent renaissance in the development of efficient algorithms for evaluating ECP integrals. In particular, multiple research groups have outlined new approaches to prescreening integrals, ([McKenzie et al., 2018](#); [Shaw & Hill, 2017](#); [Song et al., 2015](#)) greatly

reducing the computational expense. The `libecpint` library implements many of these new algorithms, combining the recursive methods and fine-grained screening of Shaw et al. (Shaw & Hill, 2017) with the higher-level screening of other recent work (McKenzie et al., 2018; Song et al., 2015). The only known implementations of the latter papers are otherwise only available in proprietary software. Therefore `libecpint` represents a necessary contribution to the wider open-source computational chemistry community. It has already been adopted by multiple packages, including Entos QCore and Serenity (Barton et al., 2020), and will be part of a future release of Psi4 (Smith et al., 2020).

## Functionalities

The core functionality of `libecpint` is the evaluation of both type 1 and type 2 integrals over ECP integrals parametrised in terms of contracted sets of primitive gaussians, as described in Shaw et al. (Shaw & Hill, 2017). The component parts divide into the following functionalities:

- a built-in library of parametrised ECPs, with generic containers for Gaussian-type ECPs;
- a highly-optimised Bessel function evaluation routine;
- screening of ECP integrals over shell pairs of orbital basis functions (McKenzie et al., 2018; Song et al., 2015), across all ECPs in a system;
- fine-grained screening of the individual type 2 integrals over primitive gaussians; (Shaw & Hill, 2017)
- recursive, automatically-generated radial integral code; (Shaw & Hill, 2017)
- adaptive quadrature for integrals not covered by the recursive routines; (Flores-Moreno et al., 2006)
- first- and second-order geometric derivatives of ECP integrals over shell pairs of gaussians.

These features can be accessed via two levels of API:

- a high-level interface, where the user provides a molecular geometry and basis set, then calls routines that return the full tensor of integrals (or integral derivatives) across all shell pairs;
- a low-level interface, where the user provides parameters and deals with the primitive integrals directly.

This allows a great deal of flexibility for different use cases, potentially allowing for users to further develop or adapt the routines themselves.

All primitive integral routines have been designed to be thread safe, allowing users to readily parallelise their calculations. In addition, there is a built in testing and benchmarking suite, allowing for efficiency comparisons both with other codes, and when testing new algorithmic developments.

## Acknowledgements

Thank you to Moritz Bensberg, Peter Bygraves, Thomas Dresselhaus, Christopher Junghans, Peter Kraus, Jan Unsleber, and Jens Wehner, for finding bugs and suggesting improvements to the initial release of `libecpint`, and often providing helpful solutions.

## References

- Barton, D., Bensberg, M., Böckers, M., Dresselhaus, T., Eschenbach, P., Hellmann, L., Klahr, K., Massolle, A., Niemeyer, N., Schnieders, D., Tölle, J., Unsleber, J. P., & Neugebauer, J. (2020). *Qcserenity/serenity: Release 1.3.1* (Version 1.3.1) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.4059510>
- Dolg, M. (2000). Effective Core Potentials. In J. Grotendorst (Ed.), *Modern Methods and Algorithms in Quantum Chemistry* (Second, Vol. 3, pp. 507–540). John von Neumann Institute for Computing. ISBN: 3-00-005834-6
- Dolg, M. (2002). Chapter 14 Relativistic effective core potentials. In *Theoretical and Computational Chemistry* (Vol. 11, pp. 793–862). Elsevier. [https://doi.org/10.1016/S1380-7323\(02\)80040-1](https://doi.org/10.1016/S1380-7323(02)80040-1)
- Dolg, M., & Cao, X. (2012). Relativistic Pseudopotentials: Their Development and Scope of Applications. *Chemical Reviews*, 112(1), 403–480. <https://doi.org/10.1021/cr2001383>
- Flores-Moreno, R., Alvarez-Mendez, R. J., Vela, A., & Köster, A. M. (2006). Half-numerical evaluation of pseudopotential integrals. *Journal of Computational Chemistry*, 27(9), 1009–1019. <https://doi.org/10.1002/jcc.20410>
- Frisch, M. J., Trucks, G. W., Schlegel, H. B., Scuseria, G. E., Robb, M. A., Cheeseman, J. R., Scalmani, G., Barone, V., Petersson, G. A., Nakatsuji, H., Li, X., Caricato, M., Marenich, A. V., Bloino, J., Janesko, B. G., Gomperts, R., Mennucci, B., Hratchian, H. P., Ortiz, J. V., ... Fox, D. J. (2016). *Gaussian 16 Revision C.01*.
- Hay, P. J., & Wadt, W. R. (1985). *Ab initio* effective core potentials for molecular calculations. Potentials for the transition metal atoms Sc to Hg. *The Journal of Chemical Physics*, 82(1), 270–283. <https://doi.org/10.1063/1.448799>
- McKenzie, S. C., Epifanovsky, E., Barca, G. M. J., Gilbert, A. T. B., & Gill, P. M. W. (2018). Efficient Method for Calculating Effective Core Potential Integrals. *The Journal of Physical Chemistry A*, 122(11), 3066–3075. <https://doi.org/10.1021/acs.jpca.7b12679>
- McMurchie, L. E., & Davidson, E. R. (1981). Calculation of integrals over *ab initio* pseudopotentials. *Journal of Computational Physics*, 44(2), 289–301. [https://doi.org/10.1016/0021-9991\(81\)90053-X](https://doi.org/10.1016/0021-9991(81)90053-X)
- Shaw, R. A., & Hill, J. G. (2017). Prescreening and efficiency in the evaluation of integrals over *ab initio* effective core potentials. *The Journal of Chemical Physics*, 147(7), 074108. <https://doi.org/10.1063/1.4986887>
- Smith, D. G. A., Burns, L. A., Simmonett, A. C., Parrish, R. M., Schieber, M. C., Galvelis, R., Kraus, P., Kruse, H., Di Remigio, R., Alenaizan, A., James, A. M., Lehtola, S., Misiewicz, J. P., Scheurer, M., Shaw, R. A., Schriber, J. B., Xie, Y., Glick, Z. L., Sirianni, D. A., ... Sherrill, C. D. (2020). Psi4 1.4: Open-source software for high-throughput quantum chemistry. *The Journal of Chemical Physics*, 152(18), 184108. <https://doi.org/10.1063/5.0006002>
- Song, C., Wang, L.-P., Sachse, T., Preiß, J., Presselt, M., & Martínez, T. J. (2015). Efficient implementation of effective core potential integrals and gradients on graphical processing units. *The Journal of Chemical Physics*, 143(1), 014114. <https://doi.org/10.1063/1.4922844>
- Wadt, W. R., & Hay, P. J. (1985). *Ab initio* effective core potentials for molecular calculations. Potentials for main group elements Na to Bi. *The Journal of Chemical Physics*, 82(1), 284–298. <https://doi.org/10.1063/1.448800>
- Werner, H.-J., Knowles, P. J., Knizia, G., Manby, F. R., Schütz, M., & others. (2019). *MOLPRO, version 2019.2, a package of ab initio programs*.