Application of Reptation Quantum Monte Carlo and Related Methods to LiH

by

Egor Ospadov

B. Sc. (Honours Physics) Department of Physics and Astronomy,

University of Waterloo, Canada

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The Faculty of Mathematics and Sciences

Department of Physics

BROCK UNIVERSITY

September 11, 2012

In presenting this thesis in partial fulfillment of the requirements for advanced degree at Brock University, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Signature: ____________________________

Department of Physics
Brock University
500 Glenridge Avenue
St. Catharines, Ontario
Canada, L2S 3A1

Date: ____________________________

Abstract

This work investigates mathematical details and computational aspects of Metropolis-Hastings reptation quantum Monte Carlo and its variants, in addition to the Bounce method and its variants. The issues that concern us include the sensitivity of these algorithms' target densities to the position of the trial electron density along the reptile, time-reversal symmetry of the propagators, and the length of the reptile.

We calculate the ground-state energy and one-electron properties of LiH at its equilibrium geometry for all these algorithms. The importance sampling is performed with a single-determinant large Slater-type orbitals (STO) basis set. The computer codes were written to exploit the efficiencies engineered into modern, high-performance computing software. Using the Bounce method in the calculation of non-energy-related properties, those represented by operators that do not commute with the Hamiltonian, is a novel work.

We found that the unmodified Bounce gives good ground state energy and very good one-electron properties. We attribute this to its favourable time-reversal symmetry in its target density's Green's functions. Breaking this symmetry gives poorer results. Use of a short reptile in the Bounce method does not alter the quality of the results. This suggests that in future applications one can use a shorter reptile to cut down the computational time dramatically.

Acknowledgements

I take this opportunity to express my gratitude to my supervisor, Dr. S. M. Rothstein, who took me under his wing and shared his tremendous knowledge with me in the field of Computational Quantum Physics and Chemistry. He helped me immensely along my journey to get to where I am today and shaped me into a better scientist, for which I am very grateful.

I thank our collaborator, Dr. W. K. Yuen, for his significant contributions during our discussions centered on the topics explored in my work. His insight proved very helpful to me. I thank the members of my supervisory committee, Dr. E. Sternin and Dr. S. K. Bose, for their valuable criticism, suggestions and ideas pertaining to my research and education. I also show appreciation to all the students in the Physics and Chemistry departments for their ongoing support and lasting friendship. I am grateful to Brock University for the financial support I received over the course of these two years. I also acknowledge SHARCNET for the immense computer resources available to me. This project relied heavily on their massive computational resources.

Finally, and most importantly, I am very grateful to my family for their ongoing support during all these years of my education. Their inspiration and comfort gave me the courage and support I required. I sincerely thank you.

Contents

List of Tables

List of Figures

Abbreviations

ADF - Amsterdam Density Functional

DMC - diffusion Monte Carlo

HF - Hartree-Fock

HT - Head-Tail

MA - Middle-Adjusted

MC - Monte Carlo

MH - Metropolis-Hastings

QA - Quartile-Adjusted

QMC - quantum Monte Carlo

QZ4P - quadruple zeta plus quadruple polarization

RQMC - reptation quantum Monte Carlo

RQMC-HT - Head-Tail reptation quantum Monte Carlo

RQMC-MA - Middle-Adjusted reptation quantum Monte Carlo

RQMC-MH - Metropolis-Hastings reptation quantum Monte Carlo

SCF - self-consistent field

STO - Slater-type orbitals

WB - Weighted Bounce

WB-HT - Head-Tail Weighted Bounce

WB-MA - Middle-Adjusted Weighted Bounce

WB-QAHT - Quartile-Adjusted Head-Tail Weighted Bounce

WB-QAS - Quartile-Adjusted symmetric Weighted Bounce

1 Introduction

The advent of quantum mechanics brought to us a superior understanding of the infinitesimal world around us. With it came a difficult problem of, and our futile attempts at, solving the Schrödinger equation associated with a large many-body system. The problem of solving such a system was first tackled by using various tricks and approximations, such as the variational method or perturbation theory, to reduce the complexity of the associated equations. While this led to more development in quantum mechanics, and our deeper understanding of it, it did not lead us to our ultimate goal of solving any possible system. With the technological progress and the creation of computers, came the ability to perform numerical calculations with swiftness and ease. Even so, it still proved a daunting task to find a direct and exact solution to the Schrödinger equation (associated with large systems) through computational means. The next natural step was to approach this problem by creating new methods of solving it, either exactly or approximately. One approach is to apply the classical Monte Carlo methods to the quantum system, giving rise to quantum Monte Carlo.

Quantum Monte Carlo (QMC) is a set of versatile algorithms used to solve a modified time-independent Schrödinger equation for a system with many particles, such as atoms or molecules. Starting with a guiding wave-function Ψ, that is used for importance sampling by guiding electrons from unimportant regions of space to important regions of space, QMC can sample the desired, unknown, putatively exact ground state wave-function Φ_0 that describes the distribution of electrons in the system. Depending on the type of the QMC algorithm used, the solution to the Schrödinger equation can yield either a mixed distribution of electrons, $\Phi_0\Psi$, or their pure distribution, Φ_0^2, albeit with a minimal bias originating from the guiding function's inexact, fixed-exchange nodes when using algorithms scalable to a large number of electrons. The type of the distribution is important, because it dictates what observables can be obtained by sampling it without bias introduced by the guiding function, Ψ.

Commonly used Diffusion Monte Carlo (DMC) algorithm produces a mixed distribution, from which only a fixed node energy of the system can be determined directly, but not any other property corresponding to an operator that does not commute with the Hamiltonian without a further modification to the algorithm. The mixed estimate of such a property $\hat{A}$ introduced in DMC, $\langle\Psi|\hat{A}|\Phi_0\rangle/\langle\Psi|\Phi_0\rangle$, yields degraded results due to its inherent large guiding function bias[1]. The actual desired result is given by $\langle\Phi_0|\hat{A}|\Phi_0\rangle$.

Reptation Quantum Monte Carlo (RQMC), originally developed by S. Baroni and S. Moroni[2], is a straight-forward and computationally efficient method used to produce a pure distribution of electrons to calculate the ground-state properties associated with operators that do not commute with the Hamiltonian, without the guiding function bias or walker population control bias, as seen in methods such as DMC[2]. The only disadvantage of RQMC, in common with DMC, is an introduction of a large, but regulated time-step bias and a small, uncontrollable bias from sampling the inexact exchange nodes of the guiding function[3]. Furthermore, a Metropolis-Hastings variant of RQMC (RQMC-MH) developed by Yuen *et al.*[4] can be used to ameliorate the failure of microscopic reversibility for the evolution of electrons in imaginary time, ensuring the convergence of the Markov chain to the intended distribution.

In addition to QMC methods, there exist other methods that are designed to solve the Schrödinger equation for a system of particles. Some of these prominent methods are Density Functional Theory (DFT), Coupled-Cluster (CC), and Configuration Interaction (CI). The disadvantages of RQMC compared to these methods are the need to include a Jastrow[5] factor to account for Coulomb correlation, guiding function and time-step biases (detailed in the paragraph above), and the statistical error associated with the results. In some cases, these methods tend to be more accurate than RQMC as well, but the increased accuracy comes with a large overhead in computational resources. Hence, the big advantage of RQMC over some of these methods is its favourable scaling. RQMC scales as $O(s^3)$, DFT scales as $O(s^3)$, CC methods scale as $O(s^7)$[6], and CI scales as $O(s^{10})$ (s being the number of electrons in the system). Since the scaling dictates the practical application of these algorithms with respect to the maximum size of the system, it is evident that RQMC has the potential to be applied to very large systems (such as large molecules, molecular clusters or proteins) on par with DFT. Whereas the CC methods are used to study atoms and small molecules, but become impractical to deal with systems containing a large number of electrons. Despite the fact that CI methods are used on both small and large systems, for large systems they require intensive CPU and IO resources. Furthermore, RQMC is an *ab initio* method while DFT, when applied to large systems, is semi-empirical in nature.

As shown by previous work in our laboratory[7], RQMC-MH is a powerful method that can be used to calculate various ground-state properties that correspond to the expectation values obtainable from the pure distribution of electrons in the system, such as one- and two- electron properties and electrical response functions. Nonetheless, in some cases, the accuracy of RQMC-MH still falls short of reported literature values and could be improved. RQMC-MH tends to underestimate most, if not all in some cases, one-electron properties relative to experimental (where applicable) and theoretical results. Henceforth, the objective of this thesis is to design methods to improve upon RQMC-MH, demonstrated by calculating one-electron properties and applied to LiH (a system that previously has been extensively studied by us[7]). To achieve this, we will consider a new method that is based on original RQMC, called Bounce[8]. We will combine RQMC-MH with Bounce and various importance sampling techniques to create new ways of sampling the target distributions for our desired properties. We will also show that using commercial software (Amsterdam Density

Functional (ADF)[9][10][11]) generated wave-functions without resorting to an optimized, explicitly correlated Jastrow[5] function will yield results in good agreement with the accuracy of previous works, albeit with a corresponding loss of efficiency.

We will study the sensitivity of these algorithms to the position of the trial electron density along the reptile, the effects of time-reversal symmetry of the propagators (Green's functions), and the consequences of varying the length of the reptile. Using these observations we will distinguish the strengths and weaknesses of each algorithm, learn their operational limits, and pinpoint the best candidate(s) to calculate the most accurate and precise one-electron properties. Doing so will allow us to address previous shortcomings of RQMC-MH that we encountered when trying to calculate one-electron properties of larger molecules and electric response properties of LiH[7] and other larger molecules.

The outline of this thesis is as follows. In the next section, we present the theory behind RQMC-MH, Bounce and their variants including all the mathematical details that describe them. Next, we will describe the theory and present the formulas used to calculate one-electron properties studied in this work. Afterwards, we will discuss our results and present our findings in the Results and Discussion section, and present an overview of our findings in the Conclusions section. The theory behind our program is described in the Appendix section, including the formulas, required computational parameters, and structure of our output. Finally, some aspects of the wave-functions, geometry of LiH, various computational parameters, and truncations used in this work are discussed in more detail in the Technical Details section.

2 Reptation Quantum Monte Carlo and Related Methods

The baseline method used in this thesis for the calculation of various properties (energy and one-electron properties) is the RQMC-MH variant of reptation quantum Monte Carlo (RQMC), developed by Yuen *et al.*[4]. The variants designed to improve the algorithm and described in detail below are: Bounce, RQMC-HT*, RQMC-MA*, WB-HT, WB-MA, WB-QAHT, WB-QAS.

The basic principle behind these methods is as follows. They attempt to find a solution to the simulated Schrödinger equation by forming a sequence of Langevin diffusion moves to construct a "reptile". Next, the reptile is modified by removing a piece at one end of the reptile and adding a new piece at the opposite end (or at the same end as in the case of Bounce and its variants). The choice of which reptile to keep (old vs. new) is done via a Metropolis decision (described below). Each of the methods demonstrated in this work have their own unique Metropolis acceptance criterion, which distinguishes the methods from one another.

2.1 Modified Schrödinger Equation in Imaginary Time

The goal of the RQMC method and every one of its variants used in this work is to successfully simulate the following modified Schrödinger equation (see the Appendix for the derivation) in atomic units:

$$-\frac{1}{2}\nabla^2 f_2 + \nabla\cdot(f_2\mathbf{F}(x)) + (E_L(x) - E_0)f_2 = -\frac{\partial f_2}{\partial t} = 0 \qquad (1)$$

where f_2 is the mixed distribution given by:

$$f_2 = \Phi_0\Psi \qquad (2)$$

$\mathbf{F}(x)$ is the drift term:

$$\mathbf{F}(x) = \frac{\nabla\Psi(x)}{\Psi(x)} \tag{3}$$

$E_L(x)$ is the local energy:

$$E_L(x) = \frac{\hat{H}\Psi(x)}{\Psi(x)} \tag{4}$$

where x is the set of $3n$ coordinates of the n electrons and E_0 is the exact (true) ground state energy of the system. The RQMC method is not sensitive to the choice of E_0.

Ψ is called a "guiding function". It is used to move electrons towards important regions of space and away from unimportant regions, thereby reducing the variance of simulated quantities. For this reason it is also called an "importance sampling function". Φ_0 is the unknown true ground-state wave-function, within the fixed node approximation, that we can sample for our desired properties, as described below.

2.2 Langevin Diffusion

In RQMC (and its variants), each snapshot of the electron distribution in an instance of imaginary time (it) is called a "scale". In the initial scale, the electrons are positioned randomly in the system. Then, Langevin diffusion is performed in succession to the n electrons for L iterations to generate a chain of scales, called a "reptile", linked in imaginary time to each other, where L is the length of the reptile; Figure 1.

Langevin diffusion is the principal method of the movement of electrons in imaginary time. These are mathematical moves that have no underlying physical significance. Langevin diffusion is used in order for the propagators (Green's functions) to be known, albeit approximately. It consists of two components: drift and diffusion:

$$x_{i+1} = x_i + \tau\frac{\nabla\Psi(x_i)}{\Psi(x_i)} + \sqrt{\tau}\chi \tag{5}$$

Here, x_i is a short-hand for the $3n$ coordinates of n electrons located in the i^{th} scale. τ is the time-step, a small fraction of the imaginary time (it) at which the calculation is performed. $\tau\frac{\nabla\Psi(x_i)}{\Psi(x_i)}$ is the drift term, and $\sqrt{\tau}\chi$ is the diffusion term. Ψ is the guiding function, and χ is a random number drawn from a standard $3n$-dimensional normal distribution.

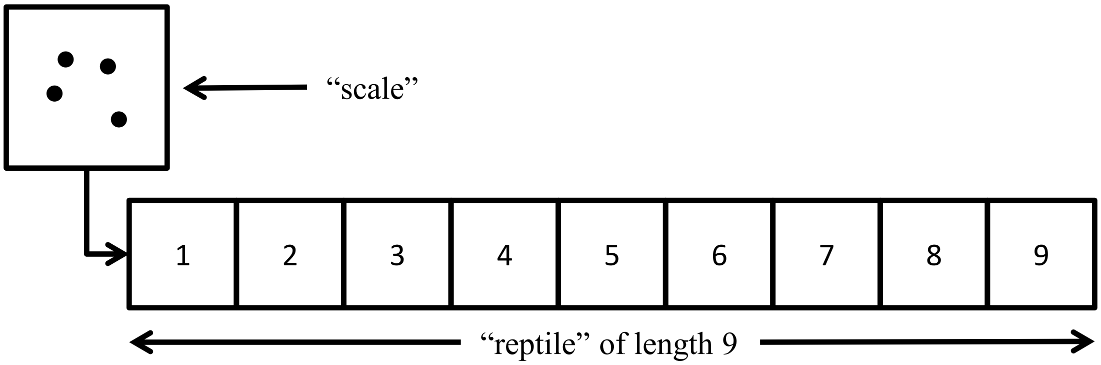

Figure 1: A reptile consisting of 9 scales. Each scale contains four electrons with a corresponding total of twelve electron coordinates. The scales are connected by drift and diffusion moves in imaginary time.

2.3 RQMC-MH

The difference between the original RQMC, presented by S. Baroni and S. Moroni[2], and RQMC-MH is that the latter, under the assumption of microscopic reversibility, allows for the reptation moves ("chops") to be performed at both the head and the tail of the reptile, and subsequently generate new scales at both the tail and the head. The former always adds to the tail after flipping the reptile with probability of 0.5.

The flipping makes Baroni and Moroni's algorithm fail the reversibility test criterion that would ensure the convergence to its target distribution. Nevertheless, Yuen *et al.* showed that despite the algorithm's failure of microscopic reversibility[4], RQMC does converge to its intended target distribution, Π:

$$\Pi(X) \propto \Psi^2(x_1)G(x_1 \to x_2)...G(x_{L-1} \to x_L)\exp(-S(X)) \tag{6}$$

where X is a reptile consisting of the following set of scales: $x_1...x_{L/2+1}...x_{L-1}x_L$. $G(x \to x')$ is a non-symmetric propagator (Green's function) for the move $x \to x'$ in imaginary time τ. $S(X)$ is the sum of the local energies across the entire reptile, given by:

$$S(X) = \tau\left[\frac{1}{2}E_L(x_1) + E_L(x_2) + \ldots + E_L(x_{L-1}) + \frac{1}{2}E_L(x_L)\right] \tag{7}$$

In RQMC-MH, the original reptile, X, is formed by Langevin diffusion. The next step, with equal probability, is to remove M scales from its head (tail) and add M new ones to its tail (head). The addition of the scales is done by performing successful Langevin diffusion moves M times starting from the tail (head) of X, thereby forming reptile Y. The action of generating Y from X is called "reptation"[12][13]; shown in Figure 2.

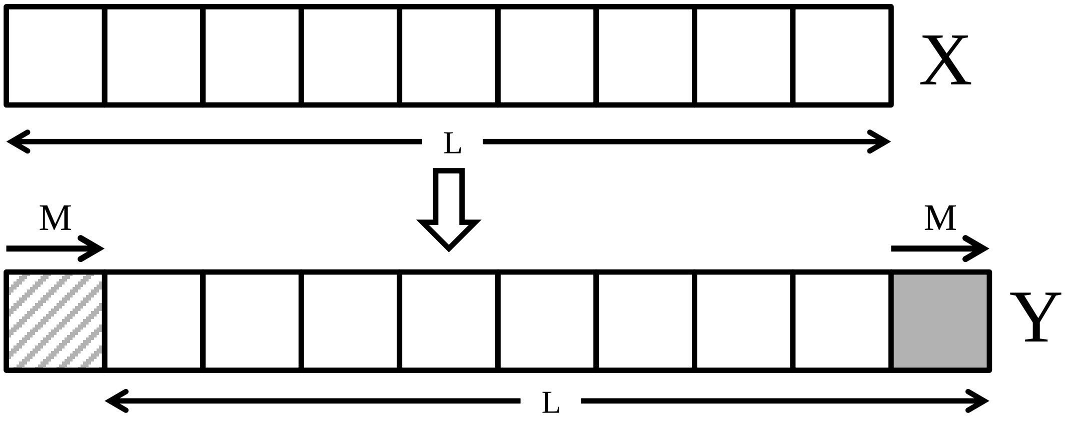

Figure 2: RQMC-MH reptation process of forming Y from X by removing M scales from the head of X and generating M scales at its tail.

Next, the choice of whether to accept the new reptile Y or reject it and keep

X is given by a simple Metropolis decision. In Metropolis-Hastings, the acceptance probability for a proposed move from X to Y with a corresponding proposal density $P(X \to Y)$ is given by:

$$A(X \to Y) = \min \left\{ 1, \frac{\Pi(Y)P(Y \to X)}{\Pi(X)P(X \to Y)} \right\} \tag{8}$$

Following the work of Yuen *et al.*[4], by assuming microscopic reversibility for the time evolution of electrons:

$$G(x \to x')\Psi^2(x) = G(x' \to x)\Psi^2(x') \tag{9}$$

the acceptance probability of the new reptile Y is simply given by:

$$A(X \to Y) = \min \left\{ 1, e^{S(X)-S(Y)} \right\} \tag{10}$$

After convergence, the algorithm generates the pure distribution, Φ_0^2, at the middle of the reptile, and the mixed distribution, $\Psi\Phi_0$, at its head and tail; as it is shown schematically in Figure 3.

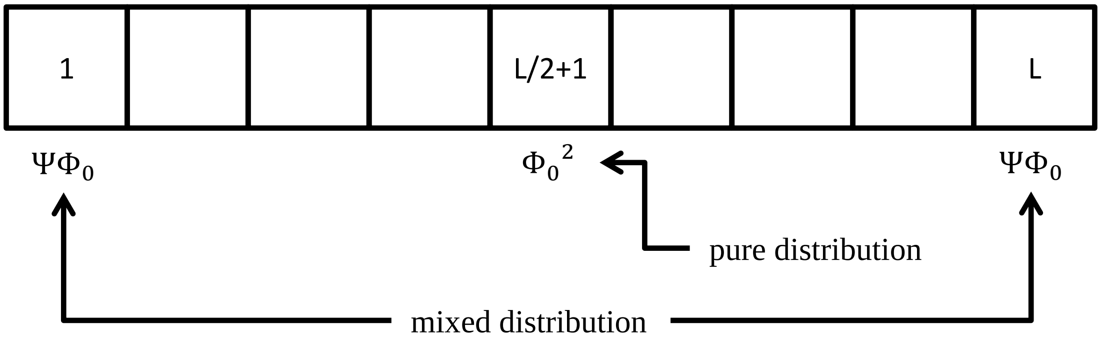

Figure 3: Distributions after a completed RQMC-MH process.

2.4 Bounce

The Bounce variant of RQMC was proposed by Pierleoni *et al.*[8]. Here, the chop-size, M, is equal to one, and one continues chopping in one direction until the trial reptile is rejected by a Metropolis decision. Then the chopping proceeds in the opposite direction. A single scale is added to the opposite side of the reptile after each chop. Pierleoni *et al.*[8] showed that the Bounce algorithm improves upon the RQMC (and hence on RQMC-MH) algorithm in a sense that it is able to produce a better energy at the head and the tail of the reptile.

The advantage of this approach is its efficient refreshment of the scales, and unlike RQMC, its target distribution, Π_B, given by Eq. (11), has desirable symmetry under time reversal.

$$\begin{aligned}\Pi_B(X) \quad &\propto \quad \sqrt{\Psi^2(x_1)G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\quad \times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})\Psi^2(x_L)}\exp(-S(X))\end{aligned} \tag{11}$$

where the propagator is the Green's function of drift and diffusion moves:

$$G(x \to x') \propto \exp\left[-\rho(x \to x')/(2\tau)\right] \tag{12}$$

and ρ is of the form:

$$\rho(x \to x') = \left| x' - x - \tau\frac{\nabla\Psi(x)}{\Psi(x)}\right|^2 \tag{13}$$

Here S(X) is the sum of the local energies accumulated on the path X given by Eq. (7), just like in the case of RQMC-MH and the local energies are given by Eq. (4).

By simplifying Eq. (11), we can write the Bounce distribution as:

$$\Pi_B(X) \propto \left|\Psi(x_1)\Psi(x_L)\right| G_s(x_1, x_2)...G_s(x_{L-1}, x_L) \tag{14}$$

where G_s is a symmetric propagator:

$$\begin{aligned} G_s(x,x') &= \exp\left[-\frac{\tau}{2}\left(E_{loc}(x)+E_{loc}(x')\right)\right]\sqrt{G(x\to x')G(x'\to x)} \\ &\propto \exp\left[-\frac{\tau}{2}\left(E_{loc}(x)+E_{loc}(x')\right)-\frac{1}{4\tau}\left(\rho(x\to x')+\rho(x'\to x)\right)\right] \end{aligned} \tag{15}$$

Furthermore, unlike RQMC, the target distribution of the Bounce algorithm is known at each instance of imaginary time: we know both distributions mathematically, but only on the average in the case of RQMC as we are moving in an arbitrary direction and chopping more than one scale at a time. In the Bounce's case, we only chop one scale and know the direction of propagation of the reptile, specifying its distribution at each instance of imaginary time. The trial move of one scale is based on the drift and diffusion in a fixed direction until the move is rejected, in which case the next move will be in the opposite direction.

Formally, when the current direction is adding to the tail, at x_L, then the new reptile $Y = x_2x_3...x_{L+1}$ is accepted with probability:

$$A_t(X\to Y) = \min\left\{1, \frac{|\Psi(x_2)\Psi(x_{L+1})|G_s(x_L,x_{L+1})G(x_2\to x_1)}{|\Psi(x_1)\Psi(x_L)|G_s(x_1,x_2)G(x_L\to x_{L+1})}\right\} \tag{16}$$

The ratio of the propagators can be simplified to:

$$\begin{aligned} \frac{G_s(x_L,x_{L+1})G(x_2\to x_1)}{G_s(x_1,x_2)G(x_L\to x_{L+1})} &= \exp[-\frac{\tau}{2}\left(E_{loc}(x_L)+E_{loc}(x_{L+1})-E_{loc}(x_1)-E_{loc}(x_2)\right) \\ &\quad -\frac{1}{4\tau}(\rho(x_{L+1}\to x_L)-\rho(x_L\to x_{L+1}) \\ &\quad +\rho(x_2\to x_1)-\rho(x_1\to x_2))] \end{aligned}$$

Similarly, if the current direction is adding to the head, at x_1, then the new reptile $Y = x_0x_1...x_{L-1}$ is accepted with probability:

$$A_h(X\to Y) = \min\left\{1, \frac{|\Psi(x_0)\Psi(x_{L-1})|G_s(x_0,x_1)G(x_{L-1}\to x_L)}{|\Psi(x_1)\Psi(x_L)|G_s(x_{L-1},x_L)G(x_1\to x_0)}\right\} \tag{17}$$

with the ratio of the symmetric propagators simplifying to:

$$\begin{aligned}\frac{G_s(x_0,x_1)G(x_{L-1}\to x_L)}{G_s(x_{L-1},x_L)G(x_1\to x_0)} &= \exp[-\frac{\tau}{2}\left(E_{loc}(x_0)+E_{loc}(x_1)-E_{loc}(x_{L-1})-E_{loc}(x_L)\right) \\ &\quad -\frac{1}{4\tau}(\rho(x_0\to x_1)-\rho(x_1\to x_0) \\ &\quad +\rho(x_{L-1}\to x_L)-\rho(x_L\to x_{L-1}))]\end{aligned}$$

2.5 RQMC-MA*

Pierleoni *et al.* did not explore any properties which are sampled from the pure distribution located in the middle of the reptile. When one is after such properties, Yuen *et al.*[4][14] suggested that it is more advantageous to sample from the Middle-Adjusted distribution rather than the usual RQMC-MH distribution. This approach moves the target density to the middle of the reptile. Therefore, we investigated using the Bounce-type moves combined with RQMC-MH and with an importance sampling technique developed by Yuen *et al.* called the No-Compromise RQMC[4] (or Middle-Adjusted RQMC[14]). We refer to the combination of the two techniques as the RQMC-MA* method. Here, the * stresses that RQMC-MH chops are performed in a Bounce-like fashion: chop one scale at a time ($M = 1$) in one direction until a Metropolis decision rejects a trial reptile. Thereafter, the direction of the chop is implemented in the opposite direction.

We know the distributions of the Bounce, Π_B given by Eq. (11), and RQMC-MA*, Π stated below in Eq. (18). Henceforth, we can implement the Bounce algorithm in a RQMC-MA* fashion by adjusting our Bounce-sampled properties with weights applied to a sample of N reptiles in the following manner:

$$\begin{aligned}\Pi(X) \;\; &\propto \;\; G(x_2 \rightarrow x_1)...G(x_{L/2+1} \rightarrow x_{L/2})\Psi^2(x_{L/2+1}) \\ &\times G(x_{L/2+1} \rightarrow x_{L/2+2})...G(x_{L-1} \rightarrow x_L)\exp(-S(X))\end{aligned} \tag{18}$$

$$W_{k,i} = \frac{\Pi(X_{k,i})}{\Pi_B(X_{k,i})} \tag{19}$$

Here, k refers to the k^{th} iteration in the Bounce algorithm and i refers to the i^{th} member of a set of N reptiles. Initially we sample the Bounce distribution at the middle of the reptile. Then we can calculate the required weights to shift from Bounce distribution (Π_B) to the RQMC-MA* distribution (Π) by taking a quotient of the two known distributions, Eq. (19). Thereby we adjust our sampled properties by using the following formula:

$$\bar{P}_k = \frac{\sum_{i=1}^{N} W_{k,i}P_{k,i}}{\sum_{i=1}^{N} W_{k,i}} \tag{20}$$

Here, $P_{k,i}$ is a property given by the Bounce distribution for the i^{th} reptile and $\bar{P}_k$ is a property in the RQMC-MA* distribution at the k^{th} iteration. The latter quantity is averaged over K iterations. The inclusion of the sum of the weights in the denominator corrects the difference in the normalization constants between $\Pi(X_{k,i})$ and $\Pi_B(X_{k,i})$.

To make sure that the first reptile represents a good starting distribution, we start with an already equilibrated RQMC-MH ensemble of N reptiles. Then we perform the Bounce method and calculate the weights required to sample in a RQMC-MA* way. The starting weight for the i^{th} reptile is calculated in the following matter:

$$W_{1,i} = \frac{\Psi^2(x_{L/2+1})}{\Psi^2(x_1)\Psi^2(x_L)} \exp\left(-\frac{R_{1,i}}{4\tau}\right) \tag{21}$$

where,

$$\begin{aligned} R_1 &= \Big[\rho(x_2 \to x_1) + ... + \rho(x_{L/2+1} \to x_{L/2}) + \rho(x_{L/2+1} \to x_{L/2+2}) + ... + \rho(x_{L-1} \to x_L)\Big] \\ &\quad - \Big[\rho(x_1 \to x_2) + ... + \rho(x_{L/2} \to x_{L/2+1}) + \rho(x_{L/2+2} \to x_{L/2+1}) + ... + \rho(x_L \to x_{L-1})\Big] \end{aligned}$$

Here and below, the k and i indicies on x are supressed for the sake of notational simplicity.

Then, at each subsequent iteration we re-compute the weights in the following efficient manner. If the previous move was rejected, we set $W_{k,i} = W_{k-1,i}$ and $R_{k,i} = R_{k-1,i}$. If it was accepted with an addition to the tail of the reptile, then:

$$W_{k,i} = \frac{\Psi^2(x_{L/2+2})}{\Psi^2(x_2)\Psi^2(x_{L+1})} \exp\left(-\frac{R_{k,i}}{4\tau}\right) \tag{22}$$

where,

$$\begin{aligned} R_k &= R_{k-1} - [\rho(x_2 \to x_1) - \rho(x_1 \to x_2)] \\ &\quad + [\rho(x_L \to x_{L+1}) - \rho(x_{L+1} \to x_L)] + 2\Big[\rho(x_{L/2+2} \to x_{L/2+1}) - \rho(x_{L/2+1} \to x_{L/2+2})\Big] \end{aligned}$$

If it was accepted with an addition to the head of the reptile, then:

$$W_{k,i} = \frac{\Psi^2(x_{L/2})}{\Psi^2(x_0)\Psi^2(x_{L-1})} \exp\left(-\frac{R_{k,i}}{4\tau}\right) \tag{23}$$

where,

$$\begin{aligned} R_k &= R_{k-1} - [\rho(x_{L-1} \to x_L) - \rho(x_L \to x_{L-1})] \\ &\quad + [\rho(x_1 \to x_0) - \rho(x_0 \to x_1)] + 2\Big[\rho(x_{L/2} \to x_{L/2+1}) - \rho(x_{L/2+1} \to x_{L/2})\Big] \end{aligned}$$

2.6 RQMC-HT*

This algorithm is analogous to the RQMC-MA*, but here the target distribution has the trial density located at the reptile's head (h) or tail (t) (depending on the direction of the addition of new scales), with the following target densities:

$$\Pi^h(X) \propto \Psi^2(x_1)G(x_1 \to x_2)...G(x_{L-1} \to x_L)\exp(-S(X)) \tag{24}$$

$$\Pi^t(X) \propto G(x_2 \to x_1)...G(x_L \to x_{L-1})\Psi^2(x_L)\exp(-S(X)) \tag{25}$$

The idea of head-tail adjusted RQMC (RQMC-HT) is to dynamically select Π^h or Π^t in the calculation of the acceptance probabilities based on where the new scales are added.

We investigate the combination of RQMC-HT and the Bounce, both of which are based on the Metropolis-Hastings algorithm. We consider the proposed move used in the Bounce algorithm, in which a trial move of one scale based on the drift and diffusion is proposed with a fixed direction until the move is rejected, in which case the next move will be of the opposite direction. Formally, when the current direction is adding to the tail of $X = x_1x_2...x_L$ at x_L, we use Π^t in Eq. (8) so that the new reptile $Y = x_2x_3...x_{L+1}$ is accepted with probability of:

$$A_t(X \to Y) = \min\left\{1, \frac{\Psi^2(x_{L+1})G(x_{L+1} \to x_L)\exp(-S(Y))}{\Psi^2(x_L)G(x_L \to x_{L+1})\exp(-S(X))}\right\} \tag{26}$$

We can simplify the ratio of the propagators to:

$$\begin{aligned}\frac{G(x_{L+1} \to x_L)\exp(-S(Y))}{G(x_L \to x_{L+1})\exp(-S(X))} &= \exp[-\frac{\tau}{2}\left(E_{loc}(x_L) + E_{loc}(x_{L+1}) - E_{loc}(x_1) - E_{loc}(x_2)\right) \\ &\quad -\frac{1}{2\tau}\left(\rho(x_{L+1} \to x_L) - \rho(x_L \to x_{L+1})\right)]\end{aligned}$$

Similarly, if the current direction is adding to the head at x_1, we use Π^h in Eq. (8)

so that the new reptile $Y = x_0 x_1 ... x_{L-1}$ is accepted with probability of:

$$A_h(X \to Y) = \min\left\{1, \frac{\Psi^2(x_0) G(x_0 \to x_1)\exp(-S(Y))}{\Psi^2(x_1) G(x_1 \to x_0)\exp(-S(X))}\right\} \tag{27}$$

with the propagators simplifying to:

$$\begin{aligned}\frac{G(x_0 \to x_1)\exp(-S(Y))}{G(x_1 \to x_0)\exp(-S(X))} &= \exp[-\frac{\tau}{2}\left(E_{loc}(x_0) + E_{loc}(x_1) - E_{loc}(x_{L-1}) - E_{loc}(x_L)\right) \\ &\quad -\frac{1}{2\tau}\left(\rho(x_0 \to x_1) - \rho(x_1 \to x_0)\right)]\end{aligned}$$

2.7 Weighted Bounce

Here, we introduce a group of algorithms, which all sample their respective distribution by taking the Bounce distribution and converting to their own distribution using a weighting scheme. The purpose of this set of methods is to move the location of the trial wave-function in the target density, while keeping the balance and symmetry of the propagators of the Bounce distribution unchanged.

2.7.1 WB-MA

This algorithm is analogous to RQMC-MA*, but now the Bounce distribution, Eq. (11), is generated and converted by weights to the Middle-Adjusted target distribution given by:

$$\begin{aligned}\Pi(X) &\propto \Psi^2(x_{L/2+1})\sqrt{G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\quad \times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})}\exp(-S(X))\end{aligned} \tag{28}$$

Since, we are sampling from the middle of the reptile via weights, we call this a Middle-Adjusted Weighted Bounce (WB-MA) algorithm. Taking the ratio of the two

Π's, the weights for WB-MA are simply:

$$W_{k,i} = \frac{\Psi^2(x_{L/2+1})}{|\Psi(x_1)\Psi(x_L)|} \tag{29}$$

Note, the weights for WB-MA are independent of whether you are adding to the reptile's head or tail. The k and i indices on x are suppressed for notational simplicity.

2.7.2 WB-HT

This algorithm is also analogous to RQMC-MA*, but now the Bounce distribution is converted by weights to the Head-Tail Adjusted target distribution. Here, we place the trial wave-function either at the head or at the tail of the reptile in its entirety. The target distribution for Head-Tail Weighted Bounce (WB-HT) depends on the direction of addition of new scales.

If we're adding to the head (h) of the reptile, the target distribution is then:

$$\begin{aligned} \Pi^h(X) \;&\propto\; \Psi^2(x_1)\sqrt{G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\quad \times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})}\exp(-S(X)) \end{aligned} \tag{30}$$

If we're adding to the tail (t) of the reptile, the target distribution becomes:

$$\begin{aligned} \Pi^t(X) \;&\propto\; \sqrt{G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\quad \times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})}\Psi^2(x_L)\exp(-S(X)) \end{aligned} \tag{31}$$

Since we know the two direction-dependent target distributions, we can formulate the weights to go from Bounce to WB-HT as follows. If adding to the head, the weight to convert from Bounce to WB-HT is:

$$W^h_{k,i} = \left|\frac{\Psi(x_1)}{\Psi(x_L)}\right| \tag{32}$$

If adding to the tail, the weight to convert from Bounce to WB-HT is:

$$W^t_{k,i} = \left|\frac{\Psi(x_L)}{\Psi(x_1)}\right| \tag{33}$$

Again, for notational simplicity, the k and i indices on x are suppressed.

2.7.3 WB-QAHT

In this algorithm, we place the target distribution one quartile away from either the head or the tail, depending on which direction we are adding, respectively. Hence, we call this the Quartile-Adjusted Head-Tail Weighted Bounce (WB-QAHT).

If adding to the head, the distribution is given by:

$$\begin{aligned}\Pi^h(X) \quad &\propto \quad \Psi^2(x_{L/4})\sqrt{G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})}\exp(-S(X))\end{aligned} \tag{34}$$

If adding to the tail, the distribution is given by:

$$\begin{aligned}\Pi^t(X) \quad &\propto \quad \sqrt{G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ &\times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})}\Psi^2(x_{3L/4})\exp(-S(X))\end{aligned} \tag{35}$$

If adding to the head, the weight to convert from Bounce to WB-QA is:

$$W^h_{k,i} = \frac{\Psi^2(x_{L/4})}{|\Psi(x_1)\Psi(x_L)|} \tag{36}$$

If adding to the tail, the weight to convert from Bounce to WB-QA is:

$$W^t_{k,i} = \frac{\Psi^2(x_{3L/4})}{|\Psi(x_1)\Psi(x_L)|} \tag{37}$$

As before, the k and i indices on x are suppressed.

2.7.4 WB-QAS

Similarily, we move the target distribution one quartile away from both the head and the tail simultaneously, but keep the symmetry and the balance of the Bounce algorithm intact. This distribution is called the Quartile-Adjusted Symmetric Weighted Bounce (WB-QAS). Mathematically, this distribution is described by:

$$\begin{aligned}\Pi(X) \quad \propto & \quad \sqrt{\Psi^2(x_{L/4})G(x_1 \to x_2)...G(x_{L-1} \to x_L)} \\ & \times\sqrt{G(x_2 \to x_1)...G(x_L \to x_{L-1})\Psi^2(x_{3L/4})}\exp(-S(X))\end{aligned} \tag{38}$$

The weight to change the properties sampled using the Bounce distribution to the WB-QAS distribution is given by:

$$W_{k,i} = \left|\frac{\Psi(x_{L/4})\Psi(x_{3L/4})}{\Psi(x_1)\Psi(x_L)}\right| \tag{39}$$

Note that the weights for WB-QAS are independent of whether you are adding to the head or the tail. The k and i indices on x are suppressed for notational simplicity.

3 One-electron Properties

3.1 Electric Moments

Electric multipole moments (dipole, quadrupole, octupole, etc) are fundamental quantities that arise from the charge distribution inside molecules and also influence their interaction with external electric fields[15]. These moments are useful because they provide an insight into the electrical and structural properties of molecules, such as charge distribution and symmetry, respectively[15]. Depending on the structure of the molecule, sometimes some (or all) of the moments are zero and sometimes they are all non-zero. For example, the dipole, quadrupole and octupole moments of the

Hydrogen atom are all zero because its electron distribution is spherically symmetric, which serves as an indication that Hydrogen is a member of the K_h symmetry group. On the other hand, Ethylene has dipole moment of zero but a non-zero quadrupole moment which is indicative that it belongs to the D_{2h} symmetry group.

In this work, we will accurately calculate all the tensor elements of the dipole, quadrupole and octupole moments for LiH using different methods presented here. The purpose behind this is two-fold. First, we verify the accuracy of our results by comparing the dipole moments we calculated against values obtained experimentally. This will allow us to test how well our theoretical methods perform against one another and how well they agree with experiment. The quadrupole and octupole moments haven't yet been determined experimentally, so we will instead compare them with values obtained through other theoretical calculations. We also exploit the tensor properties of the moments to verify our results by observing if the tensors are traceless and obey their sum rules (listed below). Secondly, we report <u>all</u> of the components of the electric moments, unlike what is standard in the literature, for the molecule in this work.

Following Buckingham's work[15], we can write the traceless dipole ($\boldsymbol{\mu}$) octupole ($\boldsymbol{\Theta}$) and quadrupole ($\boldsymbol{\Omega}$) moment tensors that describe a system of discrete charged particles as follows:

$$\begin{aligned}
\mu_\alpha &= \sum_i q_i r_{i\alpha} \\
\Theta_{\alpha\beta} &= \frac{1}{2}\sum_i q_i \left(3r_{i\alpha}r_{i\beta} - r_i^2\delta_{\alpha\beta}\right) \\
\Omega_{\alpha\beta\gamma} &= \frac{1}{2}\sum_i q_i \left(5r_{i\alpha}r_{i\beta}r_{i\gamma} - r_i^2\left(r_{i\alpha}\delta_{\beta\gamma} + r_{i\beta}\delta_{\gamma\alpha} + r_{i\gamma}\delta_{\alpha\beta}\right)\right)
\end{aligned} \tag{40}$$

Here, α, β and γ represent the Cartesian coordinates x, y and z. The summation over i indicates the summation over all the charged particles in the molecule (both nuclei and electrons). q_i is the charge of the i^{th} particle. The vector r_i represents the position of the i^{th} particle with respect to the centre of mass of the molecule.

For LiH, which is a member of the $C_{\infty v}$ symmetry group, we have the following tensor rules[16] for $\mathbf{\Theta}$ and $\mathbf{\Omega}$:

$$\begin{aligned} \Theta_{xx} &= \Theta_{yy} \\ \Theta_{zz} &= -\left(\Theta_{xx}+\Theta_{yy}\right) = -2\Theta_{xx} \\ \Omega_{xxz} = \Omega_{xzx} = \Omega_{zxx} &= \Omega_{yyz} = \Omega_{yzy} = \Omega_{zyy} \\ \Omega_{zzz} &= -\left(\Omega_{xxz}+\Omega_{yyz}\right) = -2\Omega_{xxz} \end{aligned} \tag{41}$$

Furthermore, due to the symmetry of LiH, $\mu_x = \mu_y = 0$ and the only non-zero component of the dipole moment is μ_z. In addition, the tensors $\Theta_{\alpha\beta}$ and $\Omega_{\alpha\beta\gamma}$ are symmetric under the interchange of their indices.

3.2 Hellmann-Feynman Forces

Hellmann-Feynman forces (also referred to as electric fields) are a set of intramolecular forces that are calculated by using the Hellmann-Feynman theorem. They describe the force that is exerted on each nucleus in the molecule by every other nucleus and electron. For perfect equilibrium geometry (i.e. a completely optimized wave-function), the force felt by each nucleus in the molecule should be zero. We are able to see how accurate our methods are by calculating all the Hellmann-Feynman forces for each nucleus and observing any deviations from zero. We adjust the equation reported by Neumann *et al.*[17] to account for the difference between the SCF treatment and our quantum Monte Carlo (QMC) approach and obtain the expression for the Hellmann-Feynman force given by Eq. (42).

We also examine electric field gradients, which measure the rate of change of the electric field at a chosen nucleus created by the other charges (nuclei and electrons) in the system. Again, as in the case of the Hellmann-Feynman forces, we can observe the deviations from the true values of the electric field gradients to see how good our algorithms are at estimating quantities that are located far away from the nucleus. The expression for electric field gradient is given by Eq. (43), once again by taking the one reported by Neumann *et al.*[17] and adjusting it for our QMC approach.

$$\langle q\alpha/r_A^3\rangle = -\tilde{\sum_k} \frac{Z_k\left(R_{k\alpha}-A_\alpha\right)}{\left|R_k-A\right|^3} + \sum_i \frac{\left(r_{i\alpha}-A_\alpha\right)}{\left|r_i-A\right|'^3} \tag{42}$$

$$\begin{aligned} q_A(\alpha\beta) = & - \tilde{\sum_k} \frac{Z_k\left[3\left(R_{k\alpha}-A_\alpha\right)\left(R_{k\beta}-A_\beta\right)-\delta_{\alpha\beta}\left|R_k-A\right|^2\right]}{\left|R_k-A\right|^5} \\ & + \sum_i \frac{\left[3\left(r_{i\alpha}-A_\alpha\right)\left(r_{i\beta}-A_\beta\right)-\delta_{\alpha\beta}\left|r_i-A\right|^2\right]}{\left|r_i-A\right|'^5} \end{aligned} \tag{43}$$

Here α and β represent the Cartesian coordinates x, y and z. Variable A represents a nucleus. The index k represents summation over the nuclei with the corresponding location R_k and charge Z_k. The index i represents summation over all the electrons in the system. The chevrons imply averaging over all the reptiles (10) and iterations (200 000) of the algorithm. The $\sim$ on the top of the sum indicates that any term with $r_k = A$ in the summation should be omitted. The prime ($'$) indicates that the $1/r$ type singularity has been replaced with an appropriate truncation. For more information on the truncations, see Technical Details, below.

3.3 Diamagnetic Shielding and Susceptibility

We also calculate and report diamagnetic shielding and diamagnetic susceptibility given by Eq. (44) and Eq. (45), respectively[17]. The diamagnetic shielding is calculated with respect to each nuclei, indicated by A in Eq. (44) and the diamagnetic susceptibility is calculated with respect to the centre of mass, indicated by C in Eq. (45). The calculations of paramagnetic shieldings and susceptibilities are performed over the excited states of the molecule, hence they are outside of the scope of this work which deals with ground-state properties only.

$$\langle 1/r_A \rangle = \sum_i \frac{1}{|r_i - A|} \tag{44}$$

$$\left\langle r^2 \right\rangle = \sum_i |r_i - C|^2 \tag{45}$$

Here, the index i implies summation over all the electrons in the system. The chevrons imply averaging over all the reptiles (10) and iterations (200 000) of the algorithm.

In summary, we calculate all the components of the dipole ($\boldsymbol{\mu}$), quadrupole ($\boldsymbol{\Theta}$) and octupole ($\boldsymbol{\Omega}$) moments for LiH, however we only report the non-zero components of their tensors. For the case of symmetry equivalent tensor components, for example Θ_{xx} and Θ_{yy}, we report the average of the two, as Θ_{xx} only. We calculate and report all the Hellmann-Feynman forces for both Li ($\langle q\alpha/r^3_{Li}\rangle$) and H ($\langle q\alpha/r^3_H\rangle$) atoms. In addition, we calculate all of the electric field gradients on both Li ($q_{Li}(\alpha\beta)$) and H ($q_H(\alpha\beta)$), but only report the unique non-zero tensor components in the similar fashion of the electric moments. Finally we calculate and report the diamagnetic shielding for both Li ($\langle 1/r\rangle_{Li}$) and H ($\langle 1/r\rangle_H$) nuclei and the overall diamagnetic susceptibility $\langle r^2\rangle_C$.

4 Results and Discussion

The results of RQMC-MH have been studied previously in great detail by us[7], using $L_0 = 121$, $M_0 = 20$ and $\tau = 0.012...0.002(0.002)$. Here, L_0 corresponds to the (initial) length of the reptile at the largest time-step, τ_0. Similarily, M_0 corresponds to the (initial) RQMC-MH chop-size at the largest time-step. In our previous work, we studied the behaviour of the energy (sampled at the head and the tail of the reptiles) and the one-electron properties of interest to us in this thesis (sampled in the middle of the reptile). The properties we obtained in that work were of high quality and such were obtained through heavy optimization and many CPU hours. For the purpose of this thesis, we deliberately chose a set of time-steps and reptile lengths such that the results obtained with them from RQMC-MH would likely be inferior to the results obtained from our previously published work. This was done in order to test our algorithms against RQMC-MH at a set of larger time-steps and smaller length reptiles with a purpose of finding an algorithm that gives better properties than RQMC-MH and takes less computational resources.

For the sake of computational efficiency, the time-steps used in this study were $\tau = 0.12...0.02(0.02)$, a factor of 10 larger than what we published. We investigated initial reptile lengths of $L_0 = 13$, $L_0 = 61$ and $L_0 = 121$, with corresponding initial chop-sizes for RQMC-MH being $M_0 = 3$, $M_0 = 10$ and $M_0 = 20$, respectively. For all of these explorations we kept the number of reptiles fixed at $N = 10$. We ran all the algorithms using the same random seeds to be able to compare them to each other more accurately. The length of the reptile was varied as a function of time-step[18] (as we decrease τ, we increase L), given by:

$$L(\tau) = L_0 \left(\frac{\tau_0}{\tau}\right)^{\left(\frac{3}{2}\right)} \tag{46}$$

The RQMC-MH chop-size, M, was also varied as a function of time-step:

$$M(\tau) = M_0 \left(\frac{\tau_0}{\tau}\right)^{\left(\frac{3}{2}\right)} \tag{47}$$

The results of our calculations are tabulated in Tables 1-4. We also present our results in terms of box plots in Figures 4-7 that show the relative error of our results versus known values in the literature. The box plots contain only values that have a corresponding non-zero literature value. This is done intentionally, because one cannot calculate a relative error of a property that has a literature value of zero.

Table 1: Energy and one-electron properties for RQMC-MH and Bounce

Property	RQMC-MH $L_0 = 13$	Bounce $L_0 = 13$	RQMC-MH $L_0 = 61$	Bounce $L_0 = 61$	RQMC-MH $L_0 = 121$	Bounce $L_0 = 121$	Literature
E_0	-8.081(2)	-8.076(2)	-8.082(2)	-8.0742(6)	-8.084(2)	-8.074(1)	-8.070553(5)[a]
μ_z	2.259(3)	2.31(1)	2.285(1)	2.33(2)	2.288(1)	2.26(3)	2.314(1)[b], 2.3140[c]
Θ_{xx}	1.476(4)	1.58(6)	1.527(3)	1.58(6)	1.536(2)	1.44(9)	1.5485[d]
Θ_{zz}	-2.954(7)	-3.09(4)	-3.055(2)	-3.11(15)	-3.074(4)	-2.82(5)	-3.097[c]
Ω_{xxz}	-2.66(2)	-2.74(4)	-2.77(1)	-3.05(7)	-2.777(7)	-2.34(22)	-2.75(7)[e]
Ω_{zzz}	5.32(7)	5.51(6)	5.55(2)	5.47(63)	5.59(3)	4.23(41)	5.50(14)[e]
$\langle qx/r_{Li}^3\rangle$	-0.0002(3)	0.0002(5)	-0.0002(3)	0.000(2)	-0.0006(2)	0.000(1)	0.0[f]
$\langle qy/r_{Li}^3\rangle$	-0.0003(4)	-0.0005(4)	-0.0003(4)	0.0004(8)	-0.0004(3)	-0.0014(9)	0.0[f]
$\langle qz/r_{Li}^3\rangle$	0.056(3)	0.053(2)	0.057(2)	0.054(6)	0.055(2)	0.056(2)	0.0[f]
$\langle qx/r_H^3\rangle$	-0.0003(3)	0.0005(4)	-0.0001(2)	0.000(1)	-0.00015(8)	0.000(1)	0.0[f]
$\langle qy/r_H^3\rangle$	-0.0000(1)	-0.0002(3)	-0.0002(2)	0.001(1)	-0.0001(4)	-0.001(1)	0.0[f]
$\langle qz/r_H^3\rangle$	0.0108(4)	0.017(2)	0.013(1)	0.017(4)	0.014(1)	0.012(2)	0.0[f]
$q_{Li}(xx)$	0.0208(2)	0.0201(7)	0.0206(5)	0.020(2)	0.0198(3)	0.018(1)	0.0204(1)[g]
$q_{Li}(zz)$	-0.0411(5)	-0.036(1)	-0.0407(8)	-0.039(1)	-0.041(2)	-0.034(6)	-0.0408(1)[h]
$q_H(xx)$	-0.0269(4)	-0.0308(4)	-0.0273(3)	-0.0312(4)	-0.0270(4)	-0.029(3)	-0.0246(1)[g]
$q_H(zz)$	0.0539(7)	0.055(1)	0.0551(6)	0.060(3)	0.054(1)	0.057(3)	0.0491(1)[h]
$\langle r^2\rangle$	22.52(2)	22.58(7)	22.328(9)	22.36(37)	22.323(4)	21.97(24)	22.28(4)[e]
$\langle 1/r_{Li}\rangle$	5.92(2)	5.85(4)	5.91(2)	5.85(4)	5.91(2)	5.89(4)	5.912(4)[e]
$\langle 1/r_H\rangle$	2.221(1)	2.225(4)	2.231(1)	2.239(9)	2.2281(6)	2.230(7)	2.231(2)[e]

[a] Explicitly correlated Gaussian (ECG), ref. [19].

[b] Experiment, ref. [20].

[c] Explicitly correlated coupled cluster (CCSD(T)), ref. [21].

[d] Calculated from the theoretical data for (zz).

[e] Reptation quantum Monte Carlo (RQMC-MH), ref. [7].

[f] Exact value.

[g] Calculated from the experimental data for (zz).

[h] Experiment, ref. [22].

Table 2: Energy and one-electron properties for $L_0 = 13$

Property	RQMC-MH	Bounce	RQMC-HT*	RQMC-MA*	WB-HT	WB-MA	WB-QAHT	WB-QAS
E_0	-8.081(2)	-8.076(2)	-8.049(1)	-8.083(3)	-7.982(2)	-7.984(1)	-7.984(1)	-7.964(1)
μ_z	2.259(3)	2.31(1)	2.48(1)	2.30(1)	2.323(5)	2.325(4)	2.318(5)	2.31(1)
Θ_{xx}	1.476(4)	1.58(6)	1.68(2)	1.42(5)	1.61(5)	1.83(4)	1.69(5)	1.70(5)
Θ_{zz}	-2.954(7)	-3.09(4)	-3.39(12)	-3.09(4)	-3.16(2)	-3.29(3)	-3.21(2)	-3.17(2)
Ω_{xxz}	-2.66(2)	-2.74(4)	-3.09(10)	-2.15(35)	-2.89(7)	-3.10(4)	-2.99(9)	-2.94(7)
Ω_{zzz}	5.32(7)	5.51(6)	5.98(47)	4.38(33)	5.36(69)	7.35(44)	6.21(54)	6.31(48)
$\langle qx/r_{Li}^3\rangle$	-0.0002(3)	0.0002(5)	-0.0012(6)	-0.0002(5)	0.0003(6)	0.0002(4)	0.0003(4)	0.0003(6)
$\langle qy/r_{Li}^3\rangle$	-0.0003(4)	-0.0005(4)	0.0002(5)	-0.0005(5)	-0.0006(3)	-0.0001(2)	-0.0002(6)	-0.0003(6)
$\langle qz/r_{Li}^3\rangle$	0.056(3)	0.053(2)	0.028(1)	0.054(2)	0.049(2)	0.084(3)	0.061(2)	0.064(2)
$\langle qx/r_H^3\rangle$	-0.0003(3)	0.0005(4)	-0.0005(5)	0.0001(3)	0.0003(5)	-0.0003(5)	0.00005(6)	0.00003(6)
$\langle qy/r_H^3\rangle$	-0.0000(1)	-0.0002(3)	-0.0000(4)	0.0003(3)	-0.0002(4)	-0.0003(6)	-0.0004(5)	-0.0007(7)
$\langle qz/r_H^3\rangle$	0.0108(4)	0.017(2)	0.032(1)	0.016(3)	0.022(2)	-0.029(5)	0.008(3)	0.004(4)
$q_{Li}(xx)$	0.0208(2)	0.0201(7)	0.014(1)	0.022(1)	0.0183(2)	0.0297(4)	0.0211(4)	0.0214(7)
$q_{Li}(zz)$	-0.0411(5)	-0.036(1)	-0.028(2)	-0.036(1)	-0.0319(7)	-0.0602(9)	-0.0437(9)	-0.0434(5)
$q_H(xx)$	-0.0269(4)	-0.0308(4)	-0.0369(7)	-0.028(1)	-0.0304(6)	-0.023(2)	-0.026(2)	-0.0287(9)
$q_H(zz)$	0.0539(7)	0.055(1)	0.072(1)	0.055(2)	0.058(2)	0.039(3)	0.060(2)	0.0603(8)
$\langle r^2\rangle$	22.52(2)	22.58(7)	23.96(11)	22.54(8)	23.10(7)	18.95(15)	21.48(9)	21.15(9)
$\langle 1/r_{Li}\rangle$	5.92(2)	5.85(4)	4.92(4)	6.11(1)	5.86(3)	7.94(5)	5.87(3)	5.88(3)
$\langle 1/r_H\rangle$	2.221(1)	2.225(4)	2.18(7)	2.232(3)	2.198(4)	2.66(1)	2.310(2)	2.333(2)

Table 3: Energy and one-electron properties for $L_0 = 61$

Property	RQMC-MH	Bounce	RQMC-HT*	RQMC-MA*	WB-HT	WB-MA	WB-QAHT	WB-QAS
E_0	-8.082(2)	-8.0742(6)	-8.049(2)	-8.079(2)	-7.9828(7)	-7.9883(6)	-7.9898(2)	-7.9712(5)
μ_z	2.285(1)	2.33(2)	2.48(1)	2.29(1)	2.32(2)	2.38(2)	2.36(2)	2.37(2)
Θ_{xx}	1.527(3)	1.58(6)	1.78(2)	1.42(5)	1.61(5)	1.83(4)	1.69(5)	1.70(5)
Θ_{zz}	-3.055(2)	-3.11(15)	-3.76(8)	-2.88(8)	-3.04(14)	-3.67(11)	-3.33(13)	-3.38(12)
Ω_{xxz}	-2.77(1)	-3.05(7)	-3.11(21)	-2.15(35)	-3.19(10)	-3.80(11)	-3.32(10)	-3.27(11)
Ω_{zzz}	5.55(2)	5.47(63)	6.37(29)	4.38(33)	5.36(69)	7.35(44)	6.21(54)	6.31(48)
$\langle qx/r_{Li}^3\rangle$	-0.0002(3)	0.000(2)	-0.0010(8)	0.000(2)	0.000(2)	0.0005(9)	0.000(2)	0.000(2)
$\langle qy/r_{Li}^3\rangle$	-0.0003(4)	0.0004(8)	0.000(1)	-0.0012(8)	0.0002(7)	-0.0008(5)	0.0000(7)	0.000(1)
$\langle qz/r_{Li}^3\rangle$	0.057(2)	0.054(6)	0.033(4)	0.061(5)	0.053(7)	0.097(6)	0.059(7)	0.061(7)
$\langle qx/r_H^3\rangle$	-0.0001(2)	0.000(1)	0.000(1)	0.0006(6)	-0.0001(1)	-0.001(2)	0.000(1)	0.000(1)
$\langle qy/r_H^3\rangle$	-0.0002(2)	0.001(1)	0.000(2)	-0.0009(9)	0.000(1)	0.003(1)	0.002(1)	0.002(1)
$\langle qz/r_H^3\rangle$	0.013(1)	0.017(4)	0.035(6)	0.012(3)	0.017(4)	-0.039(6)	0.014(2)	0.013(2)
$q_{Li}(xx)$	0.0206(5)	0.020(2)	0.013(2)	0.025(2)	0.019(1)	0.0360(5)	0.0211(4)	0.0217(7)
$q_{Li}(zz)$	-0.0407(8)	-0.039(1)	-0.028(2)	-0.048(1)	-0.038(2)	-0.077(2)	-0.041(3)	-0.043(3)
$q_H(xx)$	-0.0273(3)	-0.0312(4)	-0.036(2)	-0.029(2)	-0.0306(7)	-0.023(2)	-0.033(2)	-0.030(1)
$q_H(zz)$	0.0551(6)	0.060(3)	0.072(3)	0.054(3)	0.061(2)	0.037(4)	0.060(2)	0.0603(8)
$\langle r^2\rangle$	22.328(9)	22.36(37)	23.65(34)	22.21(32)	22.44(44)	17.21(19)	21.73(27)	21.49(25)
$\langle 1/r_{Li}\rangle$	5.91(2)	5.85(4)	4.93(6)	6.06(2)	5.85(4)	7.75(5)	5.87(3)	5.87(4)
$\langle 1/r_H\rangle$	2.231(1)	2.239(9)	2.20(2)	2.246(5)	2.24(1)	2.763(9)	2.26(1)	2.269(9)

Table 4: Energy and one-electron properties for $L_0 = 121$

Property	RQMC-MH	Bounce	RQMC-HT*	RQMC-MA*	WB-HT	WB-MA	WB-QAHT	WB-QAS
E_0	-8.084(2)	-8.074(1)	-8.050(1)	-8.080(2)	-7.9828(7)	-7.989(2)	-7.989(1)	-7.972(1)
μ_z	2.288(1)	2.26(3)	2.42(5)	2.25(4)	2.26(3)	2.37(2)	2.28(3)	2.28(3)
Θ_{xx}	1.536(2)	1.44(9)	1.91(2)	1.45(12)	1.44(10)	1.82(2)	1.48(7)	1.48(6)
Θ_{zz}	-3.074(4)	-2.82(5)	-3.69(27)	-2.80(7)	-2.82(5)	-3.64(6)	-2.93(9)	-2.95(10)
Ω_{xxz}	-2.777(7)	-2.34(22)	-3.38(37)	-2.40(41)	-2.40(23)	-3.44(14)	-2.35(31)	-2.31(26)
Ω_{zzz}	5.59(3)	4.23(41)	6.66(90)	4.66(92)	4.39(60)	7.00(17)	4.64(27)	4.56(26)
$\langle qx/r_{Li}^3\rangle$	-0.0006(2)	0.000(1)	-0.0002(8)	-0.001(1)	0.000(2)	0.000(2)	-0.002(2)	-0.002(2)
$\langle qy/r_{Li}^3\rangle$	-0.0004(3)	-0.0014(9)	0.0002(9)	-0.003(2)	-0.002(1)	-0.0011(5)	-0.001(1)	-0.0015(9)
$\langle qz/r_{Li}^3\rangle$	0.055(2)	0.056(2)	0.024(5)	0.061(6)	0.057(1)	0.100(4)	0.060(4)	0.061(3)
$\langle qx/r_H^3\rangle$	-0.00015(8)	0.000(1)	0.000(1)	0.001(2)	0.000(1)	-0.002(4)	0.000(2)	0.000(2)
$\langle qy/r_H^3\rangle$	-0.0001(4)	-0.001(1)	-0.0003(5)	0.000(2)	-0.002(1)	-0.001(2)	0.000(2)	0.000(2)
$\langle qz/r_H^3\rangle$	0.014(1)	0.012(2)	0.028(5)	0.004(6)	0.013(2)	-0.053(6)	0.010(4)	0.010(4)
$q_{Li}(xx)$	0.0198(3)	0.018(1)	0.017(3)	0.018(2)	0.017(1)	0.037(3)	0.020(1)	0.021(3)
$q_{Li}(zz)$	-0.041(2)	-0.034(6)	-0.037(8)	-0.04(1)	-0.034(6)	-0.079(5)	-0.036(5)	-0.038(5)
$q_H(xx)$	-0.0270(4)	-0.029(3)	-0.033(2)	-0.018(4)	-0.028(4)	-0.014(7)	-0.028(3)	-0.026(3)
$q_H(zz)$	0.054(1)	0.057(3)	0.068(5)	0.044(7)	0.056(3)	0.026(9)	0.060(4)	0.061(4)
$\langle r^2\rangle$	22.323(4)	21.97(24)	23.22(9)	21.73(13)	21.95(28)	17.15(16)	21.98(21)	21.89(16)
$\langle 1/r_{Li}\rangle$	5.91(2)	5.89(4)	5.05(8)	6.01(2)	5.89(4)	7.76(7)	5.89(4)	5.89(4)
$\langle 1/r_H\rangle$	2.2281(6)	2.230(7)	2.220(4)	2.24(1)	2.230(7)	2.754(6)	2.224(7)	2.225(6)

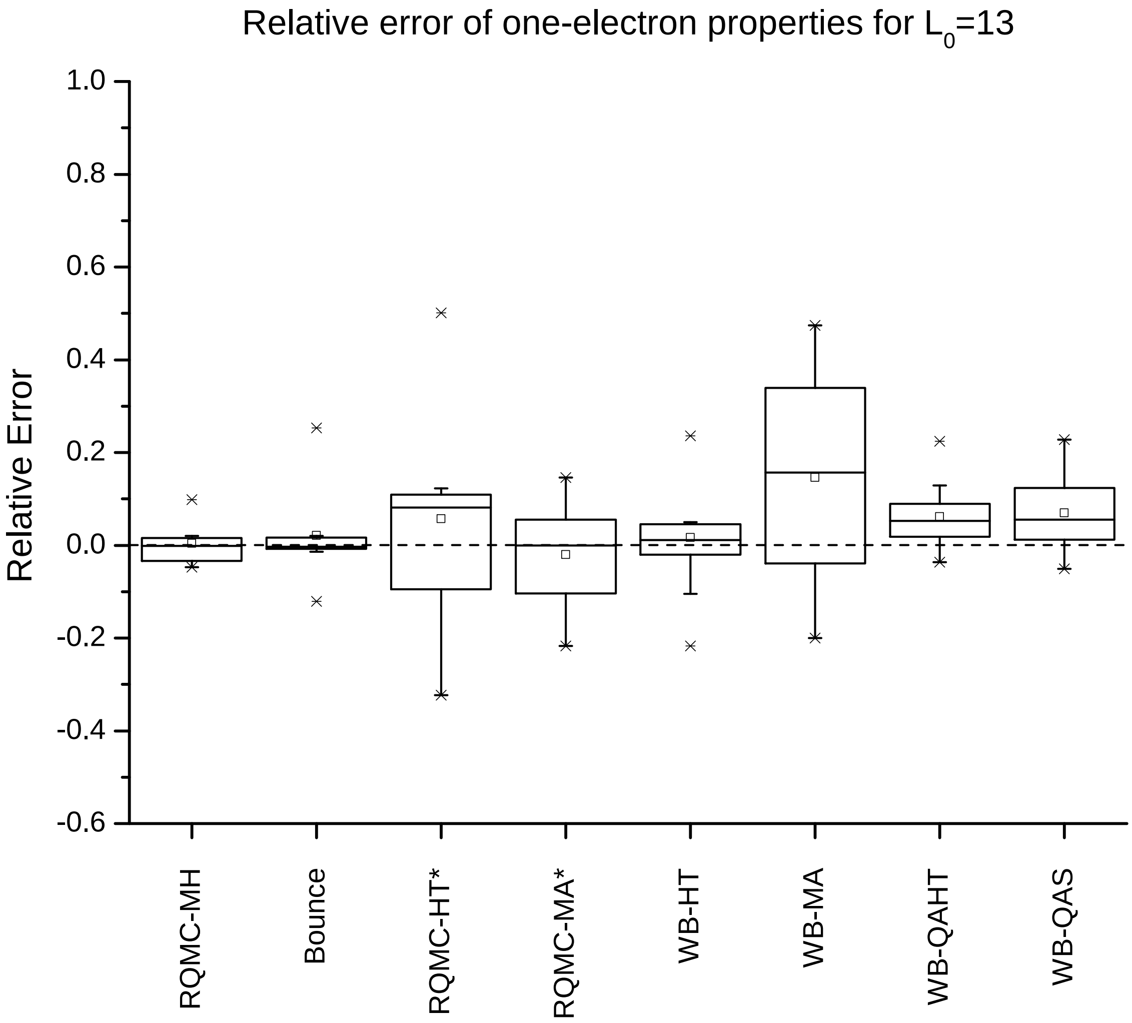

Figure 4: Box plots of relative error for all LiH properties with $L_0 = 13$. The horizontal dashed line represents zero deviation from the literature value. Here, the Bounce has the smallest relative error out of all the methods at $L_0 = 13$. The disrupting effect of placing the target distribution in the middle of the reptile or in either the head or the tail is indicated by the large boxes for RQMC-HT*, RQMC-MA*, and WB-MA*.

The horizontal dashed line that spans the entire diagram represents zero deviation from the literature value. The vertical size of the box indicates the amount of deviation our results have compared to the literature values. The bottom border of the box represents the lower quartile (25^{th} percentile), the middle line represents the median (50^{th} percentile) and the top border represents the upper quartile (75^{th} percentile). The small box within the larger box is the distribution's mean value. The top and

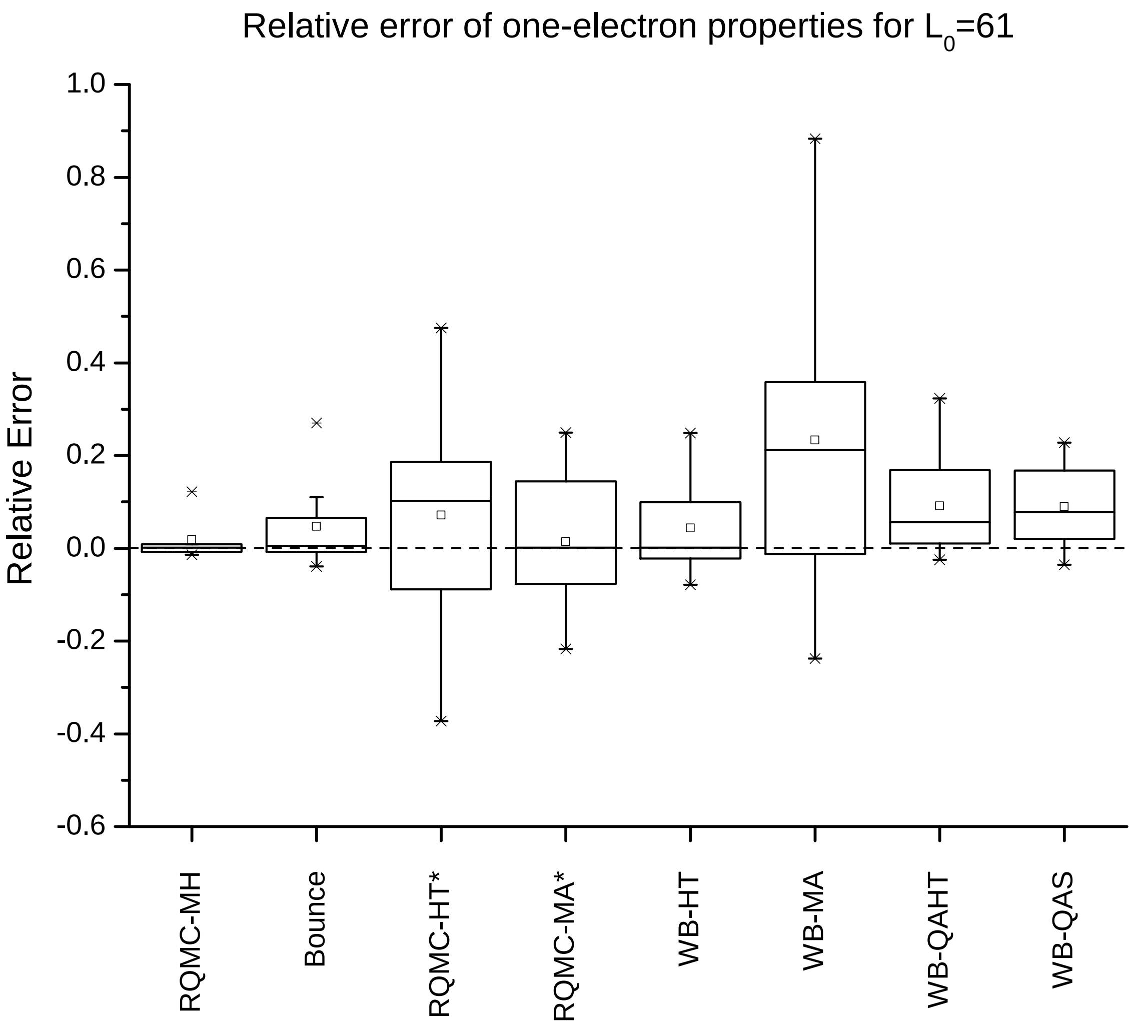

Figure 5: Box plots of relative error for all LiH properties with $L_0 = 61$. The horizontal dashed line represents zero deviation from the literature value. Here, RQMC-MH has the smallest relative error out of all the methods at $L_0 = 61$. The disrupting effect of placing the target distribution in the middle of the reptile or in either the head or the tail is indicated by the large boxes for RQMC-HT*, RQMC-MA*, and WB-MA*.

bottom whiskers, outside of the box, represent the outliers in the data. The top and bottom stars, located outside of the box, represent the maximum and the minimum of the data, respectively. Besides the box plots, we also report the energy and the dipole moment of the RQMC-MH and Bounce algorithms in Figures 8 and 9, respectively. Here, the horizontal line denotes the literature value for those quantities.

The box plots in Figures 4-6 reveal that the accuracy of the RQMC-HT* results,

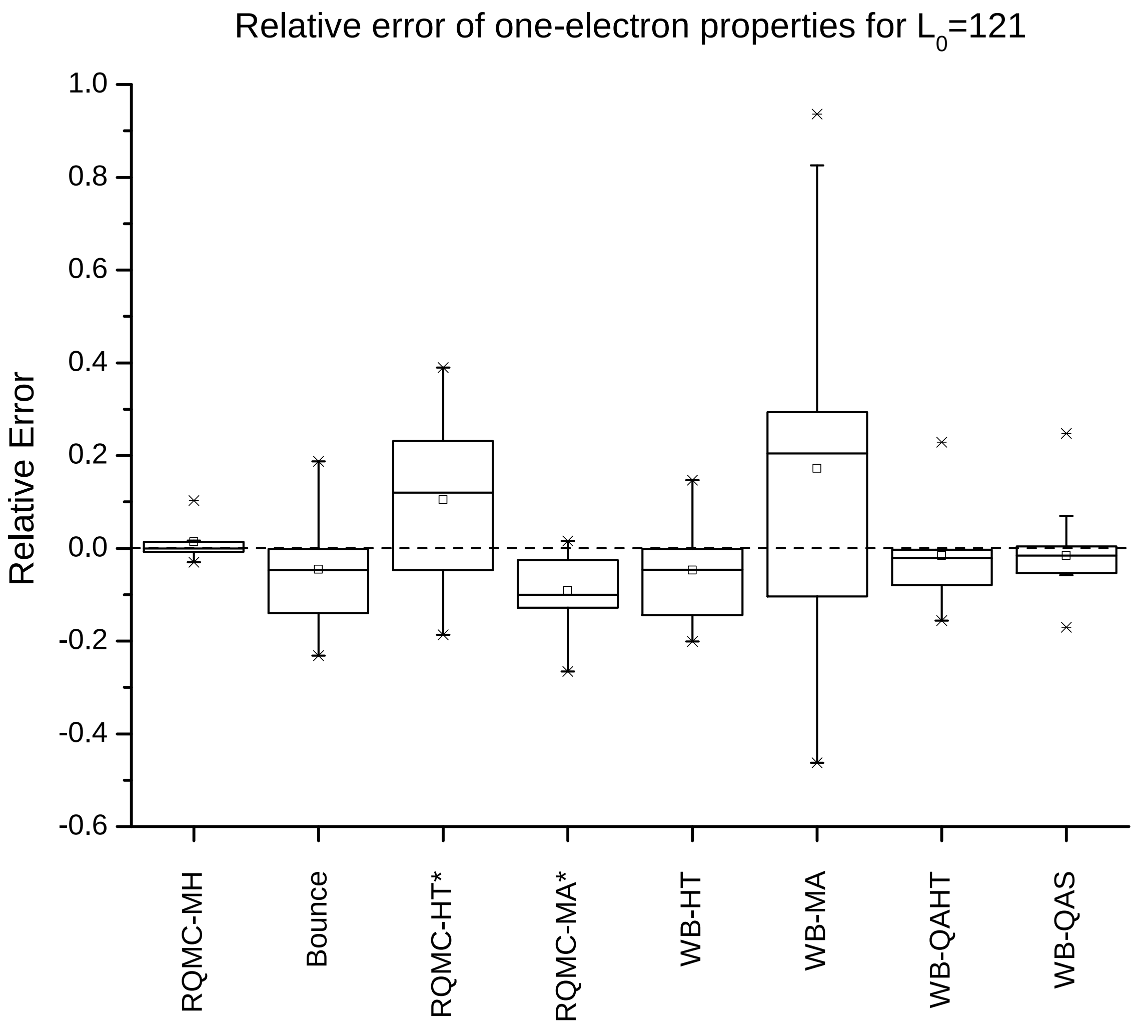

Figure 6: Box plots of relative error for all LiH properties with $L_0 = 121$. The horizontal dashed line represents zero deviation from the literature value. Here, RQMC-MH has the smallest relative error out of all the methods at $L_0 = 121$. The disrupting effect of placing the target distribution in the middle of the reptile or in either the head or the tail is indicated by the large boxes for RQMC-HT* and WB-MA*.

both the energy and one-electron properties, is much worse than that of RQMC-MH or Bounce for all the different cases of L_0. Taking a closer look at the Tables 2-4 reveals that RQMC-HT* also has worse precision than RQMC-MH or Bounce. We attribute the decrease in the quality of the results given by RQMC-HT* to the following. Our target distribution only takes the head or the tail of the reptile into account with one scale at one time (hence RQMC-HT*). Thus we effectively break the symmetry of the propagators by making their direction point to either the head,

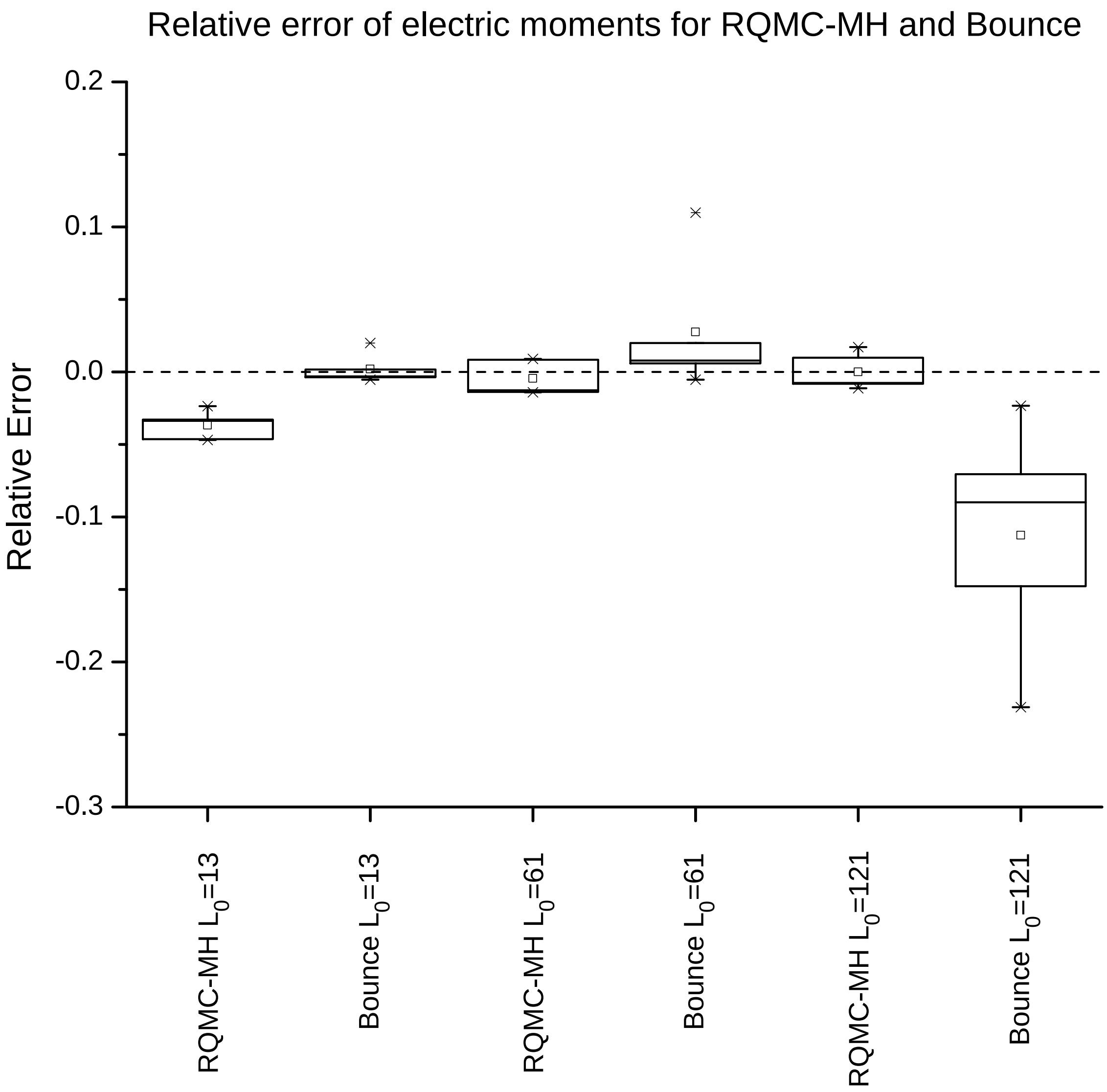

Figure 7: Box plots of electrical moments for RQMC-MH and Bounce at various L_0. The horizontal dashed line represents zero deviation from the literature value. The Bounce at $L_0 = 13$ has the smallest relative error and the highest overall accuracy, performing better than RQMC-MH at $L_0 = 121$.

Eq. (24), or the tail, Eq. (25), but not both as in the Bounce method, Eq. (11). Our results show that the Head-Tail rubric used to sample one-electron properties or energies via RQMC-HT* method is not competitive with the other algorithms.

By virtue of its employment of the trial density at the middle of the reptile, Yuen *et al.*[4] observed, for the hydrogen atom, better accuracy and less time-step bias for properties calculated using the Middle-Adjusted (MA) version of RQMC-MH. Here, we applied Bounce-type moves to RQMC-MA, denoted here as RQMC-MA*. The box plots in Figures 4-6, suggest that the accuracy of the results, both the energy and one-electron properties, given by RQMC-MA* is much worse than that of RQMC-MH or Bounce for all the different cases of L_0, with the exception of $L_0 = 121$. At $L_0 = 121$, RQMC-MA* gives slightly more accurate results than Bounce at $L_0 = 121$ only. Looking closely at the Tables 2-4 reveals that RQMC-MA* also suffers from poor precision, just like RQMC-HT* described above. The precision of RQMC-MA* is worse than RQMC-MH or Bounce for each L_0. We propose the following explanation for this. The RQMC-MA* algorithm fails because the simulated Φ_0^2 in the middle of the reptile is less balanced than the corresponding Φ_0^2 obtained in the Bounce algorithm. The Bounce algorithm is balanced better because its target distribution is more symmetric than RQMC-MA*, the head and the tail of the reptile being taken into account at each Metropolis decision, Eq. (11), rather than only the middle for the case of RQMC-MA*, Eq. (18). Therefore, we can conclude that there is absolutely no advantage in doing the RQMC-MA* over Bounce or RQMC-MH to sample the energy or one-electron properties.

Next, we look at the Weighted Bounce algorithms, where we keep the balance and the symmetry of the propagators of the Bounce unchanged, but only move the position of the trial wave-function in the target density. This gives us a better understanding of how important the position of the trial density is. The box plots, Figures 4-6, and the results given by WB-HT, Tables 2-4, suggests the following. In all cases of L_0, WB-HT gives results that are inferior to both RQMC-MH and Bounce, suffering from a loss in accuracy and precision. This suggests that evaluating the trial density in a Head-Tail fashion (either at the head or the tail of the reptile, depending on the direction of addition of new scales) is unfavourable. We attribute the decrease in the quality of the results to the breaking of the balance of the evaluation of the trial density simultaneously at the head and the tail of the reptile.

Looking at the results of the Middle-Adjusted variant of Bounce, WB-MA, we immediately note its inferior results for all cases of L_0. Both the accuracy and the precision of the WB-MA method are very poor, as indicated by the box plots in Figures 4-6 and Tables 2-4. In fact, this method seems to be the worst one out of all the methods studied in this work when it comes to both one-electron properties and the ground state energy. The failure of WB-MA can be directly attributed to the relocation of the position of the trial density to the middle of the reptile. In the Bounce method, the propagators are symmetric and their direction is towards the head and the tail of the reptile, where the trial density is located. By moving the trial density to the middle of the reptile, while keeping the direction of the propagators towards the head and the tail of the reptile, we effectively break that favourable balance and the symmetry of the Bounce method completely. Such a big unbalance in the target distribution, given by Eq. (28), is what drives the WB-MA to produce such inferior results.

The Quartile-Adjusted-Head-Tail Weighted Bounce (WB-QAHT) is just like the WB-HT method, except we shift the trial density one quartile away from either the head or the tail towards the middle, depending on the direction of the addition of new scales. By observing the box plots in Figures 4-6 and results in Tables 2-4, we note that WB-QAHT also performed worse than RQMC-MH or Bounce, but not as poorly as WB-HT or WB-MA. At $L_0 = 121$, WB-QAHT actually managed to be more accurate than Bounce, but only for one-electron properties and not for the energy. The precision of WB-QAHT is on par with Bounce but still worse than RQMC-MH. The failure of WB-QAHT can once again be attributed to the breaking down of the balance in its target distribution, Eq. (34) and Eq. (35), by placing the trial density one quartile away from either the head or the tail of the reptile. For the case of $L_0 = 121$, the apparent improved accuracy of WB-QAHT over Bounce, as seen in Figure 6, can be explained by the following. The combination of a long reptile and the placement of the trial density one quartile away from the head or the tail provide more refreshment to those scales than the middle scale, creating a favourable simulated Φ_0^2 in the middle of the reptile. This positive effect is not seen in the two other cases when $L_0 = 13$ and $L_0 = 61$, because here the length of the reptile is relatively short and the middle scale gets refreshed just as often as those containing the trial density Ψ^2.

Quartile-Adjusted Symmetric Weighted Bounce (WB-QAS) is slightly different than WB-QAHT, in that it has a more balanced trial density positioned at both the head and the tail of the reptile simultaneously. Observing the results for WB-QAS, we still see that the properties sampled from it are worse than RQMC-MH and Bounce except for the case of $L_0 = 121$, when it has a better accuracy than Bounce. The precision of WB-QAS is comparable to Bounce but worse than RQMC-MH. Just like in the case of WB-QAHT, the drop in the accuracy of the results obtained via WB-QAS can be attributed to the shift in the position of the trial wave-function from the head and the tail to one quartile away from them. Furthermore, the improved accuracy over Bounce at $L_0 = 121$ is attributed to the combination of the long reptile and the position of the trial density one quartile away from the head and the tail, just like in the case of WB-QAHT, described in detail in the previous paragraph. It is also important to note that the energy obtained from every single Weighted Bounce method was very poor. The energy was worse than RQMC-MH, Bounce, RQMC-HT* and RQMC-MA*.

Finally, we turn our attention to the results obtained from the Bounce algorithm itself. In addition to the box plots in Figures 4-6 and results in Tables 2-4, we also present the results for the Bounce and RQMC-MH for all the L_0 values in one table, Table 1, and a single box plot that contains the electrical moments, Figure 7. RQMC-MH and Bounce energies and the dipole moments are shown in Figure 8 and Figure 9, respectively. The Bounce algorithm versus RQMC-MH performed very well in terms

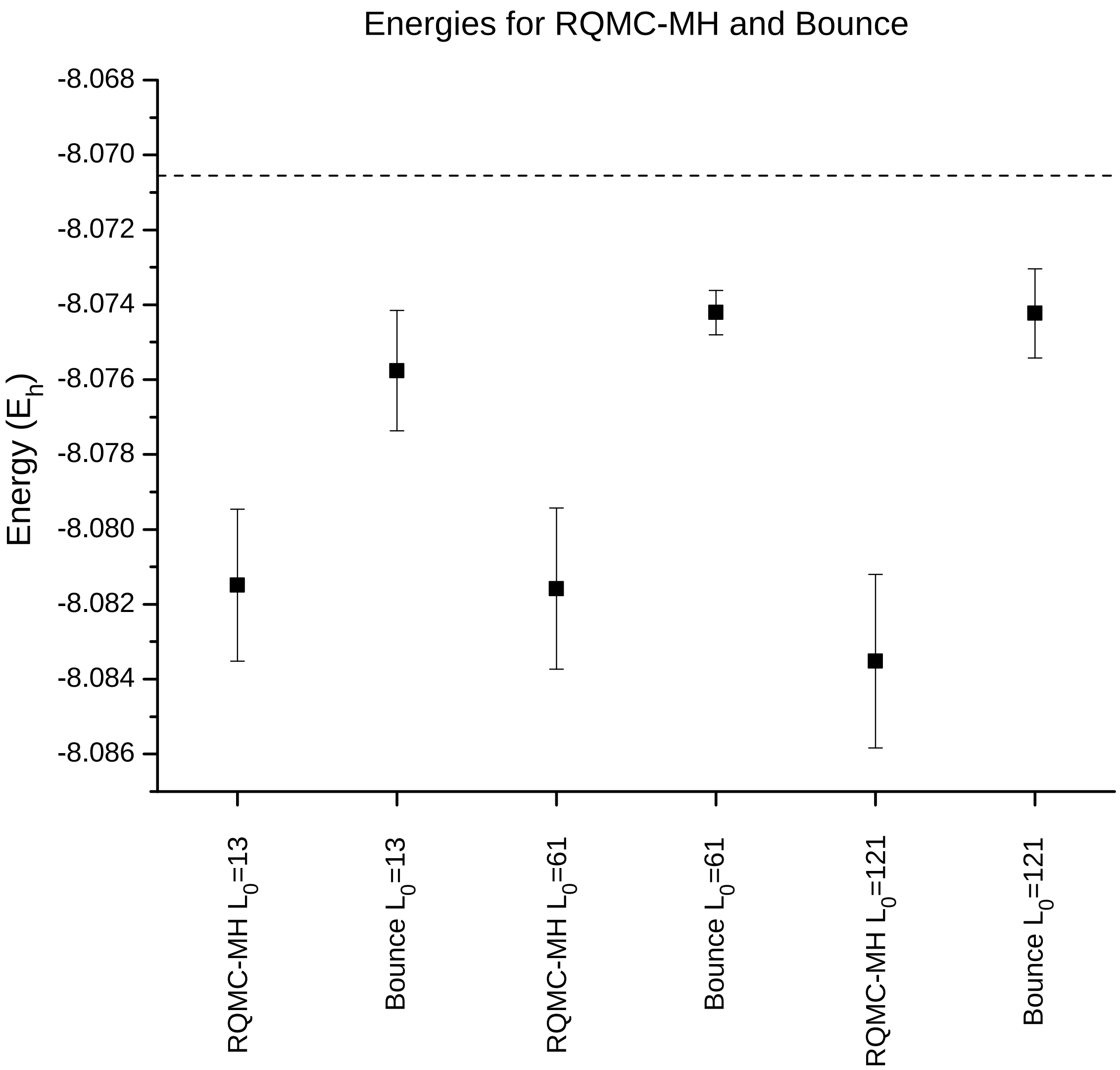

Figure 8: Energy plot for RQMC-MH and Bounce at various L_0. The horizontal dashed line represents the accepted value of the ground state energy: $E_0 = -8.070553(5)E_h$, obtained using explicitly correlated Gaussian (ECG)[19]. The Bounce at $L_0 = 61$ yields energy closest to the true ground state energy.

of the accuracy and precision. The energy sampled from the Bounce distribution is the closest to the accepted value for any choice of L_0 and has a better accuracy than RQMC-MH, as evident by Figure 8.

Examining Figure 7, which shows the relative error of selected one-electron properties for RQMC-MH and Bounce at $L_0 = 13$, 61 and 121, we note some major results. As L_0 increases, the accuracy of RQMC-MH increases, as expected, however

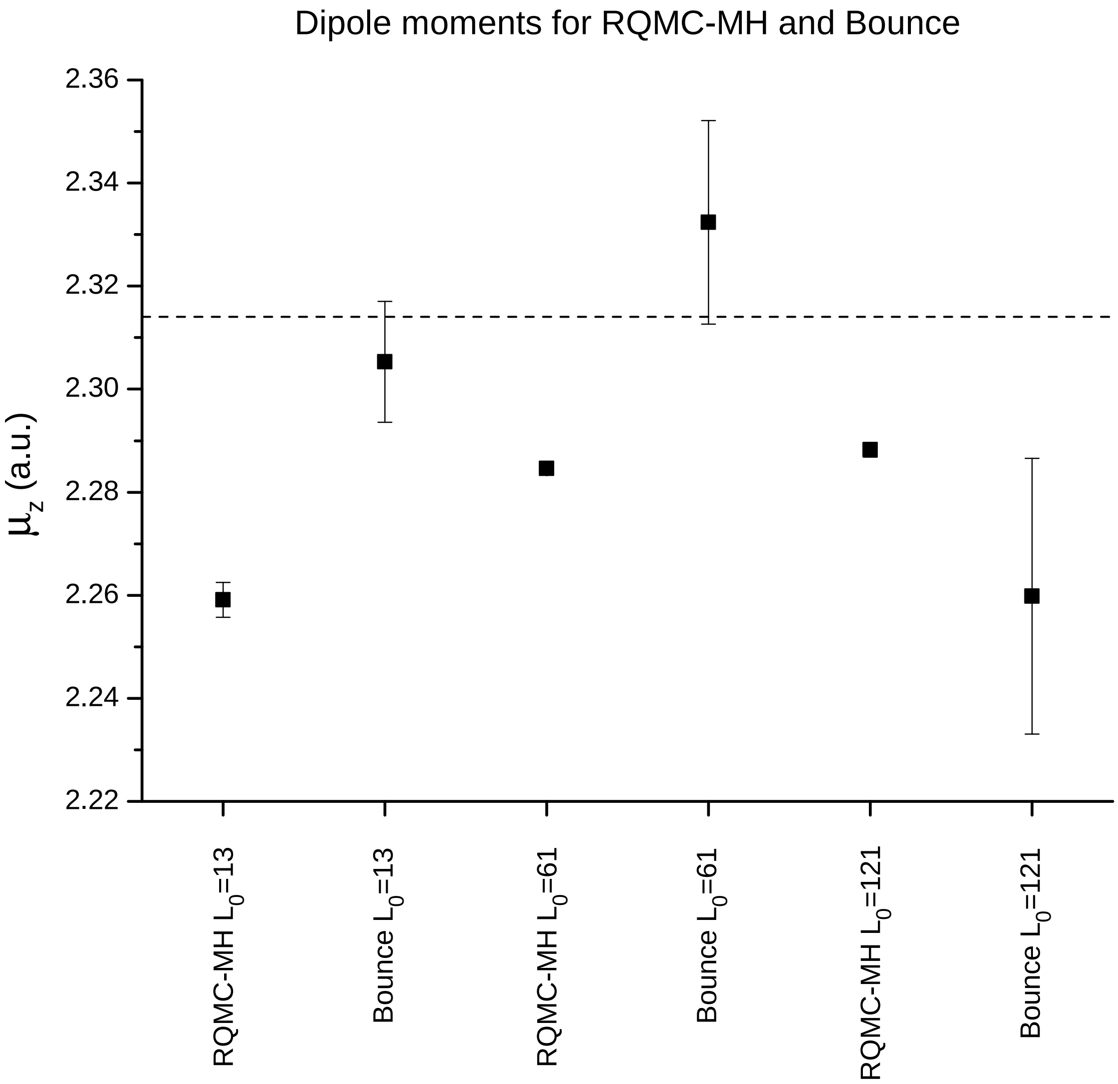

Figure 9: Plot of the dipole moment for RQMC-MH and Bounce at various L_0. The horizontal dashed line represents the accepted value of the dipole moment: $\mu_z = 2.314(1)$, obtained experimentally [20]. The Bounce at $L_0 = 13$ gives the most accurate value for the dipole moment.

the accuracy of the Bounce algorithm decreases. This is explained by the fact that RQMC-MH requires a large reptile with large chops to guarantee a desirable distribution at the middle scale. The Bounce method on the other hand needs the opposite. In its nature of chopping one scale at a time, a reptile should be as short as possible to guarantee a high refresh rate of the middle scale and thus a favourable simulated Φ_0^2 associated with it. We also note that the Bounce at $L_0 = 13$ has better accuracy than RQMC-MH at $L_0 = 121$ (essentially any RQMC-MH studied in this work) when it comes to electrical moments, Figure 7. This is further illustrated by the plot of the dipole moments in Figure 9. The Bounce at $L_0 = 13$ gives the most accurate dipole moment out of all the algorithms. The electric field gradients given by both RQMC-MH and Bounce are fairly constant and do not change with a different L_0, as shown by Table 1. However, the precision of the Bounce algorithm for these properties is found to be worse than that of RQMC-MH for all cases of L_0, as evident in Table 1.

5 Conclusions

Figures 4-6 show that placing the trial electron density in the middle of the reptile (or far away from the ends in general) degrades the results, contrary to the success of this rubric when applied to the hydrogen atom[4]. We observe that changing the symmetry of the propagators (RQMC-HT* and RQMC-MA*) also leads to a decrease in the accuracy and precision of the ground state energy and one-electron properties. The unadulterated Bounce gives the best ground state energy out of all the algorithms tested and also yields the best one-electron properties for the computationally-efficient short reptile. From this we conclude that the Bounce has favourable time-reversal symmetry in its propagators (Green's functions) and desirable balance in the position of the trial densities. Thus breaking this symmetry and balance by using the variants presented in this work gives poorer results. Sometimes, this effect is not as evident in the short reptiles because the scales at which one places the trial density are still refreshed often enough. But as one increases the reptile length, the effect is more pronounced because those scales are not refreshed as often.

By analyzing all of these algorithms, we conclude (at least with respect to LiH) that the Bounce with $L_0 = 13$ has the best accuracy for the ground state energy and one-electron properties out of all the methods considered in this study. It has favourable position of the trial electron density and time-reversal symmetry of the propagators. Performing Bounce for this short reptile provides frequent refreshment of the middle scale of the reptile at which the one-electron properties are sampled, that refreshment diminishes as the reptile gets progressively longer.

Furthermore, a short reptile decreases the computational cost associated with it dramatically. Regardless of the chosen L_0 the Bounce will always perform faster than RQMC-MH at the same L_0 due to its nature of only chopping one scale at a time. Performing RQMC-MH at $L_0 = 121$ (which represents the best RQMC-MH results in this study) takes 70 times more CPU time than Bounce at $L_0 = 13$. Hence, using Bounce at this short L_0 not only gives the best results but also saves considerable amount of CPU time. Naturally, these qualities make the Bounce algorithm a very good candidate for applications to larger molecules, which are currently in progress in our laboratory.

6 Appendix

6.1 Program Structure

In the program, the computation of the wave-function and its derivatives, the local energy and the drift is done using the method outlined by Bueckert *et al.*[23]. The distinct advantage of this method is that the derivatives are calculated analytically and not numerically. This allows for a much more efficient and precise calculation of aforementioned quantities when working with a large number of basis set parameters (atomic orbital exponents and the molecular orbital coefficients that are generated by the ADF[9][10][11] software). The precision gained from analytical differentiation is required to calculate more precise values for physical properties and for the weights used in the RQMC-MA* and Weighted Bounce variants (recall that the weights are proportional to a quotient of products of wave-functions evaluated at changed scales along the reptile). Furthermore, the use of matrix representation of the basis set parameters, wave-function and its derivatives creates more efficient and parallel code for faster execution on supercomputers. Some further modifications were applied to this method, with the following notable differences.

There are no Jastrow-type optimizations employed in the calculations, due to a part of the objective being able to demonstrate that Jastrow-type optimizations are not required when working with a sufficiently large basis set that accurately describes a system. The equation for the local energy now becomes simpler with the corresponding Jastrow terms removed.

$$E_L = -\frac{1}{2}(g^{\uparrow} + g^{\downarrow}) + U \tag{48}$$

Here, U is the potential energy function given by:

$$U = U_{en} + U_{ee} + U_{nn} \tag{49}$$

where, in obvious notation:

$$U = -\sum_i \sum_j \frac{Z_i}{|R_i - r_j|} + \sum_i \sum_{j>i} \frac{1}{|r_i - r_j|} + \sum_i \sum_{j>i} \frac{Z_i Z_j}{|R_i - R_j|} \tag{50}$$

The equations presented in this method assume that there are no unpaired electrons in the system, $N^{\uparrow} = N^{\downarrow} = N$, (Bueckert *et al*[23] does note that this is not true in general). However, our code was written in a more general way by taking into account a possible future extension of this algorithm to systems with unpaired electrons such that $N^{\uparrow} \neq N^{\downarrow}$. This would allow us to solve systems containing radicals with minimal modification of the code. The expressions for $g^{\uparrow}$ and $g^{\downarrow}$ are given by:

$$g^{\uparrow/\downarrow} = \left(\mathbf{CD}^{\uparrow/\downarrow}\right)_{\alpha\beta} \left(\mathbf{CA}^{\uparrow/\downarrow}\right)_{\beta\alpha} \tag{51}$$

The equations for drift with no Jastrow are given by:

$$\mathbf{F}_i^{\uparrow/\downarrow} = \left(\mathbf{CB}^{\uparrow/\downarrow}\right)_{\alpha i} \left(\mathbf{CA}^{\uparrow/\downarrow}\right)_{i\alpha} \tag{52}$$

Given J as the number of atomic orbitals (AO) in our basis set, $\mathbf{C}$ is a matrix consisting of $N^{\uparrow/\downarrow} \times J$ molecular orbital (MO) coefficients; $\mathbf{A}^{\uparrow/\downarrow}$ is a $J \times N^{\uparrow/\downarrow}$ matrix of atomic orbitals (χ) evaluated at the position of every electron; $\mathbf{B}^{\uparrow/\downarrow}$ is a $J \times N^{\uparrow/\downarrow} \times 3$ matrix of J gradients of atomic orbitals ($\nabla\chi$) evaluated at the position of every electron for each x, y, z component; and $\mathbf{D}^{\uparrow/\downarrow}$ is a $J \times N^{\uparrow/\downarrow}$ matrix of Laplacians of J atomic orbitals ($\nabla^2\chi$) evaluated at the position of every electron.

The $\mathbf{CA}^{\uparrow/\downarrow}$, $\mathbf{CB}^{\uparrow/\downarrow}$ and $\mathbf{CD}^{\uparrow/\downarrow}$ matrices were computed directly by including the MO coefficients in the summation over the basis set parameters. This avoids performing unnecessary matrix multiplication steps and further speeds up the computation. The equation that describes an individual atomic orbital, χ, is given by the following expression:

$$\chi = \eta r^a x^b y^c z^d exp\left(-\zeta r\right) \tag{53}$$

In conjunction with the use of the ADF-generated wave-functions, we used a different normalization constant, η, to calculate our atomic orbitals than what was used by Bueckert *et al.* The normalization constants, η, were evaluated via analytical integration of χ^2 using spherical polar coordinates, with results saved to the module file, described below. The symbol χ is not to be confused with the 3n-dimensional normal distribution function used to simulate diffusion, Eq. (5).

The program consists of three parts: a module filled with constants, a file that contains all the necessary molecular orbital parameters, and the main program. The module file contains all the parameters required to do the reptation process, Bounce algorithm and its variants, and property calculations. In particular, reptile parameters (number of scales, number of reptiles, length of each reptile, time-step, truncations, number of iterations to perform, etc.), description of the molecule (number of nuclei, charges on the nuclei, bond distance(s), location of center of mass, etc.), number of properties to calculate, names of files containing the output of the calculations, and so on. The second file contains all the molecular orbital parameters generated by the commercial software (ADF) that describe the system; The a, b, c, d, ζ coefficients, the molecular orbital coefficients $\mathbf{C}$, and our calculated normalization constants η. Finally, the main program contains all the methods which were used throughout this work. It contains functions for RQMC-MH and its variants and Bounce with its variants. It also contains the required functions for Langevin diffusion, drift, energy, wave-function calculations, and Green's functions for all the algorithms except for RQMC-MH. Furthermore it contains all the subroutines we used to calculate the one- and two- electron properties.

This design allows one to easily modify this program to calculate properties of different molecules by generating a new file for molecular orbital parameters (via

ADF) and by modifying some constants in the module. Furthermore one can reuse already created modules and parameter files to calculate more properties if those are added to the main program. The output files are broken into groups depending on the algorithm used to calculate them: RQMC-MH, Bounce, RQMC-HT*, etc. Each group has output files that contain the energy, one-electron and two-electron properties, respectively.

The execution of the program is as follows. One creates a set of module files and generates a parameter file for the system in question. Different module files will correspond to different sets of time-steps and the corresponding reptile lengths, chop-sizes, and number of reptiles. Then each program is compiled and run multiple times for a number of independent initial random seeds. After the program finishes and the output is obtained, the final quantities are obtained by the following method. First, the results from the independent runs are averaged and the standard error is computed at each time-step and plotted. Using the OriginPro[24] software package, the data is fitted using weighted polynomial regression and extrapolated to zero time-step. This gives us the values for our desired properties (energies and one-electron properties) in the limit of zero time-step, which we have tabulated in Tables 1-4. Finally, using the same software, we proceed to compile box plots and other graphs of the quantities obtained in the limit of zero time-step for further analysis.

6.2 Technical Details

In this study, the initial guiding wave-function for LiH was a single determinant HF-SCF wave-function with a QZ4P basis set generated using the ADF[9][10][11] software package. The basis set consisted of 71 Slater-type orbitals (STOs) on Li and 23 STOs on H with the corresponding SCF energy of -7.91957 E_h. The best know ground state energy of LiH is $-8.070553(5)$ E_h[19], obtained by an explicitly correlated Gaussian (ECG) calculation. The experimental bond length of LiH was taken to be 3.0154 bohr along the z-axis[25].

We did a total of 50 independent runs for each time-step. For both RQMC-MH and Bounce, the number of iterations, k, for each independent run was:

$$k = 200\,000 \tag{54}$$

We used the following set of time-steps:

$$\tau = [0.12, 0.10, 0.08, 0.06, 0.04, 0.02] \tag{55}$$

with three separate initial reptile lengths:

$$L_0 = 13, 61, 121 \tag{56}$$

The reptile lengths were varied as a function of time-step using the following relation[18]:

$$L(\tau) = L_0 \left(\frac{\tau_0}{\tau} \right)^{\left(\frac{3}{2}\right)} \tag{57}$$

In our formulation of RQMC-MH, the length of the reptile must be odd at any time-step. This is done to insure a perfectly symmetric position of the middle scale. For example, if $L = 61$, then the head is located at 1, the middle scale is located at 31, and the tail is located at 61. Hence the middle scale is exactly 30 scales away from both the head and the tail.

The corresponding initial RQMC-MH chop-sizes used were:

$$M_0 = 3, 10, 20 \tag{58}$$

The RQMC-MH chop-size, M, was also varied as a function of time-step:

$$M(\tau) = M_0 \left(\frac{\tau_0}{\tau}\right)^{\left(\frac{3}{2}\right)} \tag{59}$$

The number of reptiles, N, was fixed at:

$$N = 10 \tag{60}$$

To control the drift velocity, $F_i = \frac{\nabla\Psi(x_i)}{\Psi(x_i)}$, that can push the electron too far from a region of reasonable probability, we truncate the velocity components as follows[26]:

$$F_i = \begin{cases} F_i & \text{if } |F_i| \leq 1/\tau \\ \text{sign}[1/\tau, F_i] & \text{otherwise.} \end{cases} \tag{61}$$

The local energy, E_L, of our trial function has a singularity for when an electron is close to a nucleus, leading to large fluctuations in the simulated energy adding to the time-step bias of the Monte Carlo estimates. To ameliorate this, we truncate the local energies as follows[26]:

$$(E_L(x_i) - E_T) = \begin{cases} (E_L(x_i) - E_T) & \text{if } |E_L(x_i) - E_T| \leq 1.0/\tau \\ \text{sign}[1/\tau, (E_L(x_i) - E_T)] & \text{otherwise} \end{cases} \tag{62}$$

where E_T is an estimate of E_0.

For the Hellmann-Feynman forces, we replace a/r^3 by $\frac{5}{2}a/\epsilon^3$, whenever $r \leq \epsilon$, where a is a cartesian coordinate and $\epsilon = 2\tau^{1/3}$, a small, τ-dependent number. Similarly for the electric field gradients, we replace $(3a^2 - r^2)/r^5$ by $\frac{7}{2}(3a^2 - r^2)/\epsilon^5$. These truncations[27] reduce the variance of the simulated properties by several orders of magnitude. So as not to introduce any bias into the extrapolated values, each of above-described truncations vanish in the $\tau \to 0$ limit.

6.3 Derivation of the Modified Schrödinger Equation in Imaginary Time

To derive the modified Schrödinger equation, we write the Schrödinger equation in imaginary time (it) using atomic units[28] as:

$$\left(-\frac{1}{2}\nabla^2 + V - E_0\right)\Phi_0 = -\frac{\partial \Phi_0}{\partial t} \tag{63}$$

where $\Phi_0(R,t)$ is the exact ground-state wavefunction, $V(R)$ is the potential and E_0 is the offset ground-state energy. The (R,t) dependence is suppressed for notational simplicity. We multiply both sides by the known guiding function $\Psi(R)$[28] and suppress (R) dependence for the sake of simplicity, to get:

$$\Psi\left(-\frac{1}{2}\nabla^2 + V - E_0\right)\Phi_0 = -\Psi\frac{\partial \Phi_0}{\partial t} \tag{64}$$

$$\Psi\left(-\frac{1}{2}\nabla^2 + V - E_0\right)\Phi_0 = -\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) + \Phi_0\frac{\partial \Psi}{\partial t} \tag{65}$$

since $\Psi(R)$ is time-independent, then $\Phi_0\frac{\partial\Psi}{\partial t} = 0$, yielding:

$$-\frac{1}{2}\Psi\nabla^2\Phi_0 + V\Phi_0\Psi - E_0\Phi_0\Psi = -\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) \tag{66}$$

using $\nabla^2\left(\Phi_0\Psi\right) = \Psi\nabla^2\Phi_0 + \Phi_0\nabla^2\Psi + 2\nabla\Phi_0\cdot\nabla\Psi$, write Eq. (66) as:

$$-\frac{1}{2}\left(\nabla^2\left(\Phi_0\Psi\right) - \Phi_0\nabla^2\Psi - 2\nabla\Phi_0\cdot\nabla\Psi\right) + V\Phi_0\Psi - E_0\Phi_0\Psi = -\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) \tag{67}$$

$$-\frac{1}{2}\nabla^2\left(\Phi_0\Psi\right) + \frac{1}{2}\Phi_0\nabla^2\Psi + \nabla\Phi_0\cdot\nabla\Psi + V\Phi_0\Psi - E_0\Phi_0\Psi = -\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) \tag{68}$$

using $\nabla\cdot\left(\Phi_0\nabla\Psi\right) = \nabla\Phi_0\cdot\nabla\Psi + \Phi_0\nabla^2\Psi$ to write Eq. (68) in the following form:

$$-\frac{1}{2}\nabla^2\left(\Phi_0\Psi\right) - \frac{1}{2}\Phi_0\nabla^2\Psi + \nabla\cdot\left(\Phi_0\nabla\Psi\right) + V\Phi_0\Psi - E_0\Phi_0\Psi = -\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) \tag{69}$$

$$-\frac{1}{2}\nabla^2\left(\Phi_0\Psi\right)+\nabla\cdot\left(\Phi_0\Psi\frac{\nabla\Psi}{\Psi}\right)+\left(-\frac{1}{2}\frac{\nabla^2\Psi}{\Psi}+V-E_0\right)\Phi_0\Psi=-\frac{\partial}{\partial t}\left(\Phi_0\Psi\right) \tag{70}$$

Now, substituting the following three equations into Eq. (70):

$$f_2=\Phi_0\Psi \tag{71}$$

$$\mathbf{F}(x)=\frac{\nabla\Psi(x)}{\Psi(x)} \tag{72}$$

$$E_L(x)=-\frac{1}{2}\frac{\nabla^2\Psi}{\Psi}+V=\frac{\hat{H}\Psi(x)}{\Psi(x)} \tag{73}$$

we recover the modified Schrödinger equation in imaginary time:

$$-\frac{1}{2}\nabla^2 f_2+\nabla\cdot(f_2\mathbf{F}(x))+(E_L(x)-E_0)f_2=-\frac{\partial f_2}{\partial t}=0 \tag{74}$$

References

[1] Umrigar, C. J., Nightingale, M. P., and Runge, K. J. *The Journal of Chemical Physics* **99**, 2865–2890 (1993).

[2] Baroni, S. and Moroni, S. *Physical Review Letters* **82**, 4745–4748 (1999).

[3] Anderson, J. B. *The Journal of Chemical Physics* **65**, 4121–4127 (1976).

[4] Yuen, W. K., Farrar, T. J., and Rothstein, S. M. *Journal of Physics A: Mathematical and Theoretical* **40**, F639–F646 (2007).

[5] Jastrow, R. *Physical Review* **98**, 1479–1484 (1955).

[6] Anderson, J. B. *Quantum Monte Carlo: Atoms, Molecules, Clusters, Liquids, and Solids*, 133–182. John Wiley & Sons, Inc. (2007).

[7] Ospadov, E., Oblinsky, D. G., and Rothstein, S. M. *Physical Chemistry Chemical Physics* **13**, 8031–8036 (2011).

[8] Pierleoni, C. and Ceperley, D. M. *ChemPhysChem* **6**, 1872–1878 (2005).

[9] te Velde, G., Bickelhaupt, F. M., Baerends, E. J., Fonseca Guerra, C., van Gisbergen, S. J. A., Snijders, J. G., and Ziegler, T. *Journal of Computational Chemistry* **22**, 931–967 (2001).

[10] Fonseca Guerra, C., Snijders, J. G., te Velde, G., and Baerends, E. J. *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)* **99**, 391–403 (1998).

[11] ADF2010, SCM, Theoretical Chemistry, Vrije Universiteit, Amsterdam, The Netherlands, http://www.scm.com.

[12] Kron, A. K. *Polymer Science U.S.S.R.* **7**, 1361–1367 (1965).

[13] Wall, F. T. and Mandel, F. *The Journal of Chemical Physics* **63**, 4592–4595 (1975).

[14] Oblinsky, D. G., Yuen, W. K., and Rothstein, S. M. *Journal of Molecular Structure: THEOCHEM* **961**, 29–34 (2010).

[15] Buckingham, A. D. *Quarterly Reviews, Chemical Society* **13**, 183–214 (1959).

[16] Kielich, S. and Zawodny, R. *Chemical Physics Letters* **12**, 20–24 (1971).

[17] Neumann, D. and Moskowitz, J. W. *The Journal of Chemical Physics* **49**, 2056–2070 (1968).

[18] Langfelder, P., Rothstein, S. M., and Vrbik, J. *The Journal of Chemical Physics* **107**, 8525–8535 (1997).

[19] Cencek, W. and Rychlewski, J. *Chemical Physics Letters* **320**, 549–552 (2000).

[20] Wharton, L., Gold, L. P., and Klemperer, W. *The Journal of Chemical Physics* **33**, 1255–1255 (1960).

[21] Tunega, D. and Noga, J. *Theoretical Chemistry Accounts* **100**, 78–84 (1998).

[22] Wharton, L., Gold, L. P., and Klemperer, W. *The Journal of Chemical Physics* **37**, 2149–2150 (1962).

[23] Bueckert, H., Rothstein, S. M., and Vrbik, J. *Canadian Journal of Chemistry* **70**, 366–371 (1992).

[24] Origin, OriginLab, Northampton, MA.

[25] CCCBDB Computational Chemistry Comparison and Benchmark Database http://cccbdb.nist.gov.

[26] DePasquale, M. F., Rothstein, S. M., and Vrbik, J. *The Journal of Chemical Physics* **89**, 3629–3637 (1988).

[27] Vrbik, J., Legare, D. A., and Rothstein, S. M. *The Journal of Chemical Physics* **92**, 1221–1227 (1990).

[28] Reynolds, P. J., Ceperley, D. M., Alder, B. J., and Lester, W. A. *The Journal of Chemical Physics* **77**, 5593–5603 (1982).