Gaussian processes for choosing laser parameters for driven, dissipative Rydberg aggregates

C D B Bentley and A Eisfeld

Max Planck Institute for the Physics of Complex Systems, Nöthnitzer Strasse 38, D-01187 Dresden, Germany

E-mail: Christopher.Bentley@data61.csiro.au and eisfeld@pks.mpg.de

Received 26 June 2018, revised 24 August 2018
Accepted for publication 4 September 2018
Published 25 September 2018

Abstract
To facilitate quantum simulation of open quantum systems at finite temperatures, an important ingredient is to achieve thermalization on a given time-scale. We consider a Rydberg aggregate (an arrangement of Rydberg atoms that interact via long-range interactions) embedded in a laser-driven atomic environment. For the smallest aggregate (two atoms), suitable laser parameters can be found by brute force scanning of the four tunable laser parameters. For more atoms, however, such parameter scans are too computationally costly. Here we apply Gaussian processes (GPs) to predict the thermalization performance as a function of the laser parameters for two-atom and four-atom aggregates. These predictions perform remarkably well using just 1000 simulations, demonstrating the utility of GPs in an atomic physics setting. Using this approach, we find and present effective laser parameters for generating thermalization, the robustness of these parameters to variation, as well as different thermalization dynamics.

Supplementary material for this article is available online

Keywords: excitation transfer, quantum simulator, open quantum systems, thermalization, state preparation

1. Introduction
One often encounters a situation where the outcome of an experiment depends on many control parameters that can be varied over a large range. Usually, one is interested in achieving a particular outcome. This scenario emerges in both experimental and theoretical settings.

Achieving the target outcome can be a demanding task, in particular when the space of tunable parameters is high dimensional, and when there is no simple dependence of the outcome on the parameters. If one is only interested in the optimal outcome, then various approaches have been developed for this task (e.g. the methods in [1–3]). However, one is typically also interested in the ‘stability’ of the outcome, not (just) the parameter values that give the optimal outcome. Additionally, one would like to know the regions of parameter space where one is close to the desired outcome. We quantify ‘closeness’ to the target outcome by a cost function (or simply cost); when we speak about ‘close to the desired outcome’ we mean loosely that the result is below a certain value of the cost. Similarly, when we speak about ‘optimal’ we mean the set of parameters that gives a result that is closest to the desired outcome; yielding the minimal cost. For convenience, we will denote the manifold of the cost function over parameter space as the cost landscape.

In the ideal case one would like to know the full cost landscape: the respective values of the cost for all choices of...
the tunable parameters. However, analytic descriptions of the cost landscape can rarely be found for complex systems. Also, brute force scans of the parameter space are often precluded by the time required to perform an experiment/calculation for a single set of parameters. One powerful approach is based on Gaussian processes (GPs) [4]. GPs are not only able to search for an optimum but also provide information on the full cost landscape. This is performed by regression: a GP provides a prediction of the cost landscape by fitting known data, using Bayesian updates to prior assumptions for the model. This method can provide predictions of the full cost landscape from few data points, and has been applied to predict interatomic potentials [5, 6], and the related kernel ridge regression method has also been applied to predict properties of atoms in molecules [7, 8].

Knowledge of the predicted cost landscape also provides valuable information for choosing subsequent parameters for the experiment/simulation “well”, to use resources efficiently. This is related to reinforcement learning, where previous ‘policies of action’ inform the following action policies, with some trade-off between exploring new policies and exploiting those that have worked well previously. Reinforcement learning has been applied recently for quantum control [9–12] and even to design quantum optics experiments [13]. The idea of using known points in the cost landscape to choose subsequent parameters for evaluation is known as active learning [14]. Active learning has been applied in physical settings to produce Bose–Einstein condensates [15], and for materials design [16, 17]. Active learning is particularly useful when each simulation is computationally expensive.

Here we consider such a problem in the context of using interacting Rydberg atoms for simulating open quantum system dynamics. Rydberg systems have strong interactions, and can be optically addressed and positioned [18–20]. These properties can be exploited to simulate quantum systems [21–27]. In addition, Rydberg atoms exhibit state-changing interactions similar to molecular interactions [28, 29]. Since the parameters of Rydberg atomic systems are considerably simpler to control than the molecular counterparts, these systems are a promising setting for simulating molecular dynamics such as excitation transport in light-harvesting complexes. Features necessary for simulating molecular systems have been demonstrated using Rydberg atoms, including tunable excitation transport [30–32], non-Markovian behaviour induced in the system [33], as well as controllable thermalization [34]. We will use the same setup as in [34] (related to the setups in [31, 33]), as shown in figure 1, with a laser-driven atomic environment. Our setup has thus been used to demonstrate thermalization and flexible system dynamics, with remarkable control provided by the laser parameters. In this work, we denote the tunable laser parameters (Rabi frequencies and detunings) our laser control parameters, and parameter space is the space of possible laser control parameters.

There are key remaining questions about thermalization and thermalization dynamics to show that quantum simulation of molecular systems is possible. The first questions are about locating effective laser control parameters: which sets of parameters give rise to thermalization for a given setup? How robust are these parameters to variation? It is also important to understand the relationship between parameters and thermalization temperature: for parameters that generate thermal states, is there a smooth relationship between the parameters and thermal-state temperature? Additionally, quantum simulation requires a system with sufficient size to mimic the target system. This typically requires more than a couple of system atoms, so we also have questions about scalability: do the parameters that produce thermalization change as we scale the number of atoms in the system? If so, how? Finally, approaching a thermal state in the ‘right’ way—the same way as a target molecular system—is just as important as the thermalization itself. This leads to important questions about the thermalization dynamics: are there different sets of parameters that generate thermalization, with different thermalization dynamics? By varying parameters, what control do we have over the timescale of thermalization relative to interaction timescales within the system?

Given the significance of these open questions in our setup, our outcome of interest in this work is the thermalization of the Rydberg system atoms. Before we can approach the open questions, however, we require a method for exploring the cost landscape. Here the cost is a measure of the distance between the actual state and the target thermal state, and the cost landscape is over the laser control parameters.

In this paper, we will focus on the primary problem of exploring the cost landscape for a multidimensional parameter space, and costly simulations of our setup. This investigation will provide insights into some of the various questions we have posed to facilitate quantum simulation of open systems, and provide a methodology for approaching any of these questions.
The system and environment atoms interact via the Rydberg states of the environment atoms, \(|r\rangle\). These interactions depend on the state of the system atoms:

\[
\mathcal{H}_{\text{int}} = \sum_{n,\alpha} \tilde{V}_{n\alpha}[|n\rangle \langle r| + |r\rangle \langle n|],
\]

where \(\tilde{V}_{n\alpha} = V_{n\alpha}^{(pr)} + \sum_{m=\alpha} V_{n\alpha}^{(rr)}\) is the strength of the interaction between the system in the state \(|n\rangle\) and a specific environment atom \(\alpha\) in the Rydberg state \(|r\rangle\). The interaction between system atom \(n\) in state \(|p\rangle\) (\(|s\rangle\)) and environment atom \(\alpha\) in state \(|r\rangle\) is given by \(V_{n\alpha}^{(pr)}\) (\(V_{n\alpha}^{(rr)}\)). These pairwise interactions depend strongly on distance, such that the strongest interactions are between adjacent environment and system atoms, \(\alpha = n\).

For the given setup, we can obtain the system dynamics in the following manner. We first define an initial state for our setup that is experimentally accessible and corresponds to a localized excitation: \(\rho(0) = (|p\rangle \otimes |g\ldots g\rangle)(|p\rangle \otimes |g\ldots g\rangle)\). The state \(\rho\) is then propagated in time according to the master equation:

\[
\frac{d}{dt} \rho(t) = -i[\mathcal{H}, \rho(t)] + \sum_{\alpha} \mathcal{D}[L_{\alpha}] \rho(t),
\]

where \(\mathcal{H} = \mathcal{H}_{\text{sys}} + \mathcal{H}_{\text{env}} + \mathcal{H}_{\text{int}}\) and \(\mathcal{D}[L_{\alpha}] \rho = L_{\alpha} \rho L_{\alpha}^\dagger - \frac{1}{2} (L_{\alpha}^\dagger L_{\alpha} \rho + \rho L_{\alpha}^\dagger L_{\alpha})\) terms describe the effect of spontaneous emission on the setup. Here \(L_{\alpha} = \sqrt{\Gamma_{\alpha}} (|g\rangle \langle e| + |e\rangle \langle g|)\).

We specify the inter-atomic distances, atoms and states in the setup as in [34]. The aggregate spacing is \(d = 5 \mu m\) and the aggregate-environment atom spacing is \(\delta = 2 \mu m\). We choose the states of the aggregate atoms to be \(|p\rangle = |43p\rangle\) and \(|s\rangle = |43s\rangle\) of \(^{87}\text{Rb}\). A dimer (two system atoms) then has a lifetime of approximately 56 \(\mu s\) [35], which is much longer than the timescale of dynamics that we will consider. For the environment atoms we choose the states \(|g\rangle = |5s\rangle\), \(|e\rangle = |5p\rangle\) and \(|r\rangle = |38s\rangle\) of \(^{87}\text{Rb}\). The decay rate from \(|e\rangle\) is \(\Gamma_{p} = 6.1\ \text{MHz}\).

In this setup, we want to obtain thermalization of the system. This provides a resource for quantum simulation of systems in real (thermal) environments. We are thus interested in preparing a thermal state:

\[
\rho_{\text{T}}^\theta = \frac{1}{Z} \sum_{\alpha} e^{-E_{\alpha}/(kT_{\text{eff}})} |\varphi_{\alpha}\rangle \langle \varphi_{\alpha}|,
\]

where \(T_{\text{eff}}\) is the effective temperature, \(k\) is the Boltzmann constant, and \(Z = \text{Tr}(e^{-\mathcal{H}_{\text{sys}}/(kT_{\text{eff}})})\). The eigenstates and eigenenergies of \(\mathcal{H}_{\text{sys}}\) (equation (1)) are denoted by \(|\varphi_{\alpha}\rangle\) and \(E_{\alpha}\), respectively.

Note that the temperature scale \(kT_{\text{eff}}\) is not given by an ambient temperature of a thermal bath. The ambient temperature is typically on the order of \(\mu K\), and we also have an additional atomic component of our environment. Here the temperature scale is given in terms of the interaction strength \(W\), which determines the eigenenergies \(E_{\alpha}\).

We set a time \(t_f\) for which thermalization should have happened to a given thermal state. As addressed in the introduction, control over the thermalization timescale and the temperature of the target thermal state are important aspects for simulating general open systems. In this work, we focus on cost landscape prediction and analysing the results. We
thus fix the thermalization timescale to be faster than a given \( t_p \), where the target thermal state has a given temperature \( kT_{\text{eff}} \). However, our approach can be extended by varying the values \( t_p \) and \( kT_{\text{eff}} \). We will comment on the choices of \( t_p \) and \( kT_{\text{eff}} \) in section 3.

In [34], it was shown that for the given setup, thermalization to a tunable temperature can be achieved for particular choices of the laser driving parameters \((\Omega_p, \Omega_s, \Delta_p, \text{and } \Delta_s)\). Here, we are similarly interested in investigating thermalization of the system by controlling the laser driving parameters for the environment atoms. However, unlike in [34], in this paper we are interested in knowing the cost landscape generally (how well thermalization can be performed over the full parameter space). This in turn equips us to answer other physical questions about the setup (e.g. robustness of thermalization to parameter variations or scaling the system size), as described in the introduction.

2.2. Quantifying the outcome

We want to quantify how well we prepare the target thermal state. To do this, we apply the trace distance \( D(\rho_S(t_f), \rho_{\text{th}}^{\text{rb}}(t_f)) \) to measure the distinguishability of a given state of the system \( \rho_S(t_f) \) from the target thermal state \( \rho_{\text{th}}^{\text{rb}} \):

\[
D(\rho_S(t_f), \rho_{\text{th}}^{\text{rb}}(t_f)) = \frac{1}{2} \text{Tr}(|\rho_S(t_f) - \rho_{\text{th}}^{\text{rb}}|),
\]

(6)

with \( |\rho| = \sqrt{\rho^\dagger \rho} \). The state \( \rho_S(t_f) \) is obtained by time-propagation of our setup \( \rho(t) \) (equation (4)), followed by tracing out the environment atoms. We will use \( D(\rho_S(t_f), \rho_{\text{th}}^{\text{rb}}) \) (denoted by \( D \) for convenience) as our cost function.

Note that the cost function is based on a state comparison at a single point in time, \( t_f \). We assume that the cost quantifies how well states have thermalized to the target state. However, the cost does not distinguish whether the propagated state is still changing in time; such a state could dynamically pass ‘close’ to the target state without being effectively thermalized. In section 3 of the supplemental material available online at stacks.iop.org/JPB/51/205003/mmedia, we show that states that are not yet steady have a minimal effect in our case, validating our association of the cost with effective thermalization. We also provide an alternate cost function in the supplemental material that could be used in cases where this transience issue arises.

2.3. Gaussian processes

In this paper, we numerically investigate the cost landscape. The challenge is that simulations of our setup are computationally costly. Also, an analytic model for the cost landscape has not been found for our setup. The ideal case—knowing the full cost landscape—is thus impractical even for a Rydberg dimer system (just two system atoms).

Using machine learning, we can gain insight into the full cost landscape through GP regression and prediction. GP regression fits known points in the landscape (parameters with known associated costs) and can then predict the full cost landscape. This prediction of the cost landscape attributes a Gaussian predictive distribution for the cost at any given ‘test’ parameter set. From this distribution, the predicted mean cost and standard deviation can be extracted. A functional form is assumed for the covariance of predicted costs over parameter space. The covariance can depend on length-scales for characteristic variation in the cost as a function of each parameter, and these length-scales can also be estimated to provide the best fit of the cost landscape by the GP regression.

The predicted cost landscape can be used within a numerical routine to balance optimization (landscape exploitation) with landscape exploration. We are most interested in sets of parameters providing a low cost (close to 0). However, we are not just interested in the minimal cost and its associated parameters. We want to know every low-cost parameter region, along with its extent in parameter space. This information requires an exploration of the cost landscape, with higher ‘priority’ of exploration given to regions that may have low cost values. As explained in the supplemental material (section 4), the predicted cost landscape can be used to identify likely low-cost regions of the cost landscape. Similarly, the standard deviation for the predicted cost landscape can identify regions of parameter space that should be explored due to a lack of knowledge (uncertainty in the predicted costs, such that the costs could be low). In this way, we obtain a numerical routine that uses previous runs (simulations and GP regression) to guide the parameters for subsequent simulations. We use the optimization package MLOOP [15], based on the GP regression algorithm (2.1) from [4] and implemented in scikitlearn [36].

2.3.1. Performance measures.

We would like quantitative measures of how well the GP-based numerical routine performs. We will consider measures that compare GP-predicted regions of the cost landscape with exact calculations of these same cost-landscape regions by scanning parameter space (extrinsic measures). These scans are only possible for limited regions of the cost landscape, and for small system sizes. Thus, we will also consider measures that only depend on the (region of the) cost landscape predicted by the numerical routine (intrinsic measures). In both cases, we evaluate 2D cross-sections of the cost landscape.

2.3.1.1. Extrinsic measures.

The extrinsic measures allow us to analyse the accuracy of the predicted landscape: how well it matches the exact landscape. We introduce three measures to quantify this: (a) inaccuracy, (b) inaccuracy for \( D < C_i \), and (c) penalty.

(a) The inaccuracy measure is the absolute difference between the scanned cross-sections and the predicted cross-sections. The absolute difference is taken point-wise, then averaged over every point in the cross-sections. Note that the costs obtained by both parameter scans and predictions are discrete samples from the cost landscape.

(b) The inaccuracy for \( D < C_i \) follows the same procedure as (a), but only points where the scanned cross-sections have trace distance \( D \) below some low-cost threshold \( C_i \).

3 MLOOP source code and documentation http://m-loop.readthedocs.io
are included in the average. This measure thus quantifies accuracy for the low-cost regions of the cost landscape, with the choice of \( C_i \) discussed in section 4.1.

(c) The penalty is a measure for how closely the real cost landscape fits within the standard deviation for the predicted cost landscape. The contribution to the penalty for a given cost-landscape point is the absolute distance between the real cost and the standard-deviation interval about the predicted cost (this distance is 0 when the real point is within the interval). To calculate the penalty, we take the sum of these point-wise contributions over every point in each cross-section, then average over the cross-sections.

It is important to note that we would like to apply the routine in a regime where such extrinsic measures are not possible. The main benefit of using GPs to predict the landscape is that this prediction can occur where scans are not feasible: in such cases, we will need to rely on intrinsic measures.

2.3.1.2. Intrinsic measures. We introduce four intrinsic measures: (d) imprecision, (e) imprecision for \( D < C_i \), (f) absolute convergence and (g) best cost.

(a) The imprecision is the standard deviation averaged over every point in the cross-sections.

(b) The imprecision for \( D < C_i \) follows the same procedure as (d), but only points where the predicted cost cross-sections have trace distance \( D < C_i \) are included in the average.

(c) The absolute convergence is the absolute difference between the current and previous predicted cost landscapes (averaged over points). This measure is a function of the number of runs performed by the numerical routine. A single run of the numerical routine involves selection of parameters to test, and one simulation (see section 4 of the supplemental material for details). We calculate the predicted cost landscape (cross-sections) to evaluate the performance measures every 20 runs, for numerical convenience. The ‘previous’ predicted cost landscape is thus calculated using 20 fewer runs than the ‘current’ predicted cost landscape.

(d) The best cost is the lowest cost that has been obtained: it is the optimal cost from the set of known (evaluated) points of the cost landscape.

3. Scanning parameter space for a dimer

The naive approach to obtain the cost landscape is to simply scan the variable parameters. Recall that we have four such parameters: \((\Omega_p, \Omega_c, \Delta_p, \Delta_c)\), which means an enormous number of calculations.

As a reference for the GP approach of the following sections we perform such a scan for the simplest nontrivial setup where a scan is tractable: a Rydberg dimer.

We choose the target thermal state temperature to be given by \( kT_{\text{eff}} = 1.2 \text{ W} \). This temperature has no special significance for the thermal state, except that it is not a limiting case of nearly zero or infinite temperature (in these cases all population is in the lowest eigenstate, and the eigenstates are equally populated, respectively). The methods we present can be applied for any temperature.

In addition, we require that this thermal state is reached before a certain time, which we choose here to be \( t_f = 2 \mu s \). Again, as for the choice of the temperature this time is chosen arbitrarily (but is much smaller than the lifetime of the Rydberg states of the system (\( ~56 \mu s \) [35])). As discussed in section 2.1, by choosing a time \( t_f \) we set the maximal timescale of thermalization. We propagate the state for \( 2 \mu s \) and compare the resulting state with the reference thermal state as described in section 2.2.

Even though we look at just two system atoms, a full scan of the cost landscape is not feasible. Therefore in figure 2 we present exemplary 2D cross-sections of the 4D parameter space. We choose a low-cost common point for these cross-sections: \((\Omega_p, \Omega_c, \Delta_p, \Delta_c) = (37.1, 40.1, -26.6, -74.7) \equiv \vec{x}_{\text{comm}}\), which has cost \( D = 6 \times 10^{-5} \). This low-cost point was obtained using differential evolution optimization [2]; this approach does not estimate the cost landscape. The ranges of the laser parameters are chosen to be experimental achievable [37, 38], with detunings small enough to avoid unwanted resonances.

Each of the 2D cross-sections in figure 2 are composed of \( 100 \times 100 \) points (i.e. we evaluated 10 000 sets of parameters by time-propagating the state). This number of points allows us to resolve features in parameter space up to one or two MHz. The computational time for each point in a given 2D
parameters and dynamical behaviour, one would first like to have information on the full four-dimensional parameter space.

We want to emphasize again that due to the computational cost, this (scanning) approach is not scalable to larger systems, and nor can it be extended generally to explore the larger 4D parameter space even for the Rydberg dimer.

4. Predicting the cost landscape using GPs for a dimer

We have seen that the computational cost is a fundamental challenge for numerical investigation of thermalization in our setup (and also more generally for simulation of scalable quantum systems). We thus want to obtain as much information as possible from as few simulations as possible. The information that we desire is a balance of optimization and landscape exploration: we are interested in the (multiple) regions of parameter space that give rise to low-cost thermal states, including the breadth of these regions.

4.1. GP prediction

We use GPs to predict the cost landscape for the Rydberg dimer setup. As outlined in section 2.3, GP regression fits known points in the cost landscape, and this fit can then be exploited to predict the unknown cost values of different parameters. To choose the evaluated parameters (the ‘known’ points that are fit in the GP regression), we apply a numerical routine that uses a GP to both explore the cost landscape, and exploit the landscape by focusing on low-cost regions as explained in section 4 in the supplemental material. To display, for example, a 2D cross-section of the predicted landscape, ‘test’ parameters within this cross-section (100 × 100 points in our case) are evaluated using the GP fit to the cost landscape. The output is then the predicted values for each point in the cross-section, along with the standard deviation at each point.

In figure 4 we show the predicted landscape and its associated standard deviation after 1000 runs of our numerical routine (i.e. 1000 sets of parameters were simulated). To compare directly with the numerical scans of the parameter space, we display the 2D cross-sections through the same point $\hat{x}_{\text{conn}}$ as the scans in figure 2. This comparison shows that qualitatively, the landscape prediction is very good, i.e. it identifies almost all low-cost regions. This is remarkable since just 1000 simulations have been performed. The four laser control parameters are allowed to vary freely within the ranges shown in the cross-sections; the parameters can be sampled from the full 4D parameter space and are not restricted to lie within the displayed cross-sections. This is in contrast to the 100 × 100 points for each cross-section displayed for the parameter scans, where each of the 10000 points lies within the displayed cross-sections.

As explained above, GP regression not only provides a cost-landscape but also acts as an optimizer by searching for the lowest cost. The minimum cost found by the GP has
$D < 0.001$, for parameters $(\Omega_p, \Omega_c, \Delta_p, \Delta_c)/(2\pi)$ (MHz) $= (18.9, 92.4, 9.3, 42.5)$. This set of parameters is quite far away from the common point of our calculated cross-sections, at $(37.1, 40.1, -26.6, -74.7)$. Since the ‘optimization’ part of the GP ‘explores’ the vicinity of the minimum, we expect that if we would compare cross-sections through the minimum found by the GP we would have similar or even better agreement for the predicted landscape.

In figure 4, we show the predicted landscape after 1000 runs. This number of runs is chosen based on the following considerations. Firstly, in our implementation, prediction is more expensive for a GP with more completed runs. Note that this issue can be somewhat mitigated by using matrix inversion methods that scale better with the number of runs $n$ (the matrix size is $n^2$), in place of the Cholesky factorization that we apply. A more comprehensive reduction of the computational cost for large run numbers can be achieved by performing GP regression using a subset of the completed runs (approaches for choosing such a subset are detailed in [4]). Secondly, the standard deviation of the predicted cost landscape typically decreases with the number of runs, see e.g. figure 5. There is thus a trade-off between the computational resources required for a GP-based numerical routine, and the precision of the predicted cost landscape. For the dimer setup, 1000 runs is well within the ‘sweet spot’: firstly, it is computationally much cheaper to use GPs than to scan parameters to investigate the cost landscape. Secondly, the predicted landscape is very close to the actual landscape after 1000 runs: the typical inaccuracy $< 0.1$, as shown in figure 5. Similarly, considering an intrinsic measure, the standard deviation in the predicted cost is sufficiently small to identify likely low-cost regions of the cost landscape. As seen in figure 4, after 1000 runs, the standard deviation for the predicted cross-sections typically takes a value between 0.2 and 0.3 for the low-cost regions. Note that the ‘sweet spot’, where there is both accurate landscape prediction and a computational resource reduction from scans, grows as each simulation becomes computationally more expensive (for more details, see section 4 in the supplemental material).

We now specify our choice of threshold $C_l$ for low-cost regions of the landscape. We have noted that the standard deviation for the predicted cross-section is $\sim 0.2$ to 0.3 for regions with cost $D$ below 0.4, as seen in figure 4. It is thus prudent to choose a value for $C_l$ close to this standard deviation value: then we include regions with predicted low cost, where the standard deviation for the prediction is around the size of the distance from zero cost (perfect thermalization). This way, the intrinsic likelihood (from the predicted cost) of a low cost guides our low-cost threshold. By focusing on regions of the cost landscape with a predicted cost below 0.2, we can rule out vast regions of the cost landscape after 1000 runs. This is demonstrated by the contours in the predicted-cost cross-sections of figure 4 (left), and we expect arbitrary cross-sections to have smaller low-cost regions (the cross-sections in figure 4 are through a particularly low-cost point). We thus set the ‘low-cost’ threshold $C_l = 0.2$.

4.2. Quantifying prediction performance

To provide a quantitative analysis of the GP performance, we apply the extrinsic and intrinsic performance measures defined in section 2.3.1. Each of these measures (aside from the best cost) is averaged over the six 2D cross-sections produced by varying the four laser parameters (pairwise) about the common point.

Note that the numerical routine is stochastic, so the resulting landscape prediction, as in figure 4, varies for each instance of the numerical routine. To quantify how well the GP-based numerical routine predicts the cost landscape, we have performed 100 instances of the numerical routine. In figure 5, the mean over the instances is plotted for each performance measure, along with the standard deviation.
One would expect that with more runs, i.e. more samples of the cost landscape, the landscape prediction becomes more accurate, more precisely known and approaches convergence (the landscape predictions become more similar as the number of samples increases). In figure 5, this expectation is validated: the measures are almost always improving with the

**Figure 5.** Performance measures for the six 2D cross-sections (see figures 1 and 2 of the supplemental material) have the same common point as in figures 2 and 4. These measures are defined in section 2.3.1. The upper three measures displayed are extrinsic, comparing the predicted cost landscape cross-sections with the scans. The lower four measures are intrinsic, using just the predicted cost landscape cross-sections and their standard deviations. The measures are calculated using 100 instances of the (stochastic) numerical routine, each of which were used to generate six 2D cross-sections of the predicted cost landscape and the associated uncertainty. We calculated the performance measures for each instance, and display the mean (solid lines) and the mean plus/minus the standard deviation (dashed lines) of the performance measures from the resulting measure distributions. Note that for the log-scale plots, the standard deviation below the mean may be outside the plot range.
number of runs. The inaccuracy (for $D < C_p$) decreases with the number of runs to a final value $\sim 0.05$ ($\sim 0.04$), which means that this is the average distance between the predicted and exact landscapes for all points (points with $D < 0.2$) after 1000 runs. The imprecision and imprecision for $D < C_p$ measures in figure 5 demonstrate poor initial predictions of the landscape, where the standard deviation is under-estimated prior to around 100 runs, most noticeably for the $D < C_p = 0.2$ regions. Then, after around 100 runs, the imprecision measures decrease with the number of runs (the precision improves for the predicted landscape). As expected, the absolute convergence also demonstrates increasing consistency in the predicted landscape as the number of runs increases.

The improvement (lowering) in the imprecision measures, along with the improvement (lowering) in inaccuracy, reflects successful landscape exploration by our numerical routine. The best cost also improves with the number of runs: this demonstrates the optimization aspect of our numerical routine.

Recall that the penalty grows when actual cost values lie outside the predicted standard deviation in the predicted cost. In figure 5, the penalty decreases as the prediction improves, then increases fractionally with the number of runs. The main change in this measure is the decrease in the penalty due to the improving landscape prediction (and more accurate standard deviation). Initially the prediction and its standard deviation are poor due to very little information from few previous samples of the cost landscape. The penalty measure improves significantly within 100 runs. The subsequent slight increase in the penalty could be due to narrow features in the landscape that have not been sampled. The penalty for these features would become worse as the standard deviation in these incorrect regions of the predicted landscape is reduced with the number of runs.

If we did not have access to the scanned landscape, we would only have the intrinsic measures. It is thus encouraging to note that the imprecision measure (which averages the standard deviation in the cost landscape) is an upper bound for the inaccuracy of the landscape. This means that knowing the imprecision allows us to conservatively estimate the landscape inaccuracy, at least for the Rydberg dimer setup. The standard deviation predictions are dependent on the choice of the covariance function for the GP (as well as the optimized fitting parameters within the covariance function). The covariance function describes how the predicted landscape can change away from the known points in the landscape. We have seen that the squared-exponential covariance function employed by our routine (see section 4 of the supplemental material) provides conservative standard deviation predictions. We thus expect the same behaviour when we consider larger systems using the same covariance function. Then the imprecision could be used as a conservative estimate of the inaccuracy for general system sizes.

From another intrinsic measure, the absolute convergence measure, it appears that we could set a certain convergence threshold as a criteria to stop the numerical routine. That is, when the predicted landscapes change less than a given amount between subsequent predictions, we might expect that the routine has converged ‘close’ (as determined by the threshold) to the correct cost landscape. This approach is limited, however, when we consider individual instances of the numerical routine. Although in figure 5 the absolute convergence measure (almost) monotonically decreases with the number of runs for 100 routine instances, figure 6 for the quadromer demonstrates that a single instance involves much more fluctuation in this measure. This is due to the stochastic nature of both the sampling and the GP regression to fit the known points in the cost landscape.

The results that we have presented are for a particular choice of numerical routine, which we have found to work well for our setup. This choice is explained in detail in section 5 of the supplemental material, along with alternative routines.

5. Predicting the cost landscape using GPs for a quadromer

We have seen that scanning parameter space does not provide an approach that can be scaled to larger system sizes within a reasonable computational time. However, the prediction procedure using GPs performed qualitatively remarkably well within a much shorter time. We now apply this approach to gain insights from the cost landscape of a Rydberg quadromer system.

We use our numerical routine to predict 2D cross-sections of the cost landscape with the common point $x_{\text{comm}}$ as for figures 2 and 4, for comparison. These cross-sections, obtained for the quadromer, are presented in figure 7. Here, as for the predicted cost landscape for the dimer in figure 4, 1000
The results are shown all six are shown in section 1 of the supplemental material. Two cores common point (four of the cross-sections are shown in figure 7, all six are shown in section 1 of the supplemental material). The results are shown (as a function of the number of runs) in figure 6. Since the performance measures here are calculated using a single instance of the numerical routine, rather than 100 instances as in figure 5, we observe much more fluctuation. Nonetheless, it can be observed from figure 6 that there is an improving trend for each measure, which is similar to the trend in figure 5 for the dimer setup.

In figure 6, the imprecision drops to $\sim 0.23$ after 1000 runs. Thus, we again set the 'low-cost' threshold $C_l = 0.2$. The final value of the imprecision for $D < C_l$, which is typically lower than the imprecision, is $\sim 0.16$. As for the dimer, we expect that the imprecision measures provide a conservative upper bound on the inaccuracy of the predicted landscape (this relates to the functional form of the covariance, discussed in section 4.2 for the dimer, and is also consistent with our samples of the quadromer cost landscape compared with the corresponding GP predictions). Thus, we expect the predicted cost landscape for the quadromer to have inaccuracy $< 0.23$ overall, and we expect the inaccuracy for $D < C_l = 0.2$ measure to be less than 0.16.

The best cost in figure 6 reaches a value of $\sim 4 \times 10^{-3}$ after 300 runs and remains at this value. This is a little higher than the mean best cost over 100 instances for the dimer case ($\sim 1.5 \times 10^{-3}$). Nonetheless, this best cost value demonstrates that very-low-cost ($D \sim 0.01$) regions exist for the quadromer setup. These can also be seen in the predicted cost landscape and standard deviation for which selected cuts are shown in the supplemental material (figure 4).

Achieving a target imprecision for $D < C_l$ could be used as a stopping criterion for the numerical routine (such that no more runs are performed), as it is a useful low-cost identification measure and has a relatively smooth dependence on the number of runs. In contrast, the absolute convergence fluctuates dramatically, which would make this measure difficult to apply as a stopping criterion.

As we did for the dimer, we can now use the predicted (quadromer) cost landscape to provide a preliminary investigation of how or whether different parameters give rise to
different thermalization dynamics. In figure 7, as well as in figure 4 in the supplemental material (which is centered at the best cost), different low-cost regions are identified in the predicted cost landscape. We sampled points from these regions and we show their dynamics in figure 8. The four eigenstate populations undergo different dynamics, which is shown with an inset zoomed in on the initial dynamics for clarity. The predicted costs for each point are provided with the actual costs in the figure, and the differences in each case are much lower than the standard deviation in the predicted cost (as is also the case in almost every point that we validated from the quadromer landscape). While the populations in the subplots (a), (c) and (d) become steady very close to the desired eigenstate populations, the dynamics in (b) are not as close (with higher cost). Nonetheless, this point in the cost landscape (which also has a faster timescale to reach a steady-state) could be used to perform local optimization and locate a nearby lower cost with similar associated dynamics.

6. Conclusions

We have tackled the general problem of extracting much information from few costly simulations. We demonstrated that GPs perform admirably at this task in the setting of quantum simulation in atomic physics.

The successful application of GPs also provided physical insight into various questions required for simulating molecular systems. In particular, we identified sets of parameters that give rise to thermalization, as well as the robustness of these parameter regions. Using the parameter space information provided by GPs, we demonstrated that different thermalization dynamics can be observed in our setup. We also obtained preliminary information about how parameters that result in thermalization vary as the system size changes. Our method provides a useful approach for further study of this relationship. Similarly, one could vary the temperature of the target thermal state, and observe the resulting changes in the low-cost regions of parameter space. Our approach thus supports the development of physical insight into the controllability of our setup for molecular simulation.

It is important to note that the cost landscape could be scanned experimentally, since the experiment duration is just 2 μs (independent of the system size), and we are interested in a parameter space described by experimentally tunable laser parameters. Our investigations in this paper confirmed that our setup does provide key elements required for molecular simulation, and added to the quantum simulation toolbox by
identifying low-cost parameter regions, robustness, and different thermalization dynamics. This provides a foundation and motivation for experimental exploration of the setup.

Acknowledgments

The authors thank Jan-Michael Rost for fruitful discussions. We acknowledge funding from the DFG: grant EI 872/4-1 through the Priority Programme SPP 1929 (GiRyd).

ORCID iDs

A Eisfeld @ https://orcid.org/0000-0001-9645-440X

References

[1] Boyd S and Vandenberghe L 2004 Convex Optimization (Cambridge: Cambridge University Press)
[2] Storn R and Price K 1997 Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces J. Global Optim. 11 341–59
[3] Dueck G and Scheuer T 1990 Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing J. Comput. Phys. 90 161–75
[4] Rasmussen C E and Williams C K I 2006 Gaussian Processes for Machine Learning (Cambridge: MIT Press)
[5] Bartók A P, Payne M C, Kondor R and Csányi G 2010 Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons Phys. Rev. Lett. 104 136403
[6] Bartók A P and Csányi G 2015 Gaussian approximation potentials: a brief tutorial introduction Int. J. Quantum Chem. 115 1051–7
[7] Rupp M, Tkatchenko A, Müller K-R and von Lilienfeld O A 2012 Fast and accurate modeling of molecular atomization energies with machine learning Phys. Rev. Lett. 108 058301
[8] Rupp M, Ramakrishnan R and von Lilienfeld O A 2015 Machine learning for quantum mechanical properties of atoms in molecules J. Phys. Chem. Lett. 6 3309–13
[9] Bukov M, Day A G R, Sels D, Weinberg P, Polkovnikov A and Mehta P 2017 Reinforcement learning in different phases of quantum control arXiv:1705.00565v2
[10] August M and Hernández-Lobato J M 2018 Taking gradients through experiments: LSTMs and memory proximal policy optimization for black-box quantum control arXiv:1802.04063v2
[11] Fösel T, Tighineanu P, Weiss T and Marquardt F 2018 Reinforcement learning with neural networks for quantum feedback arXiv:1802.08267v2
[12] Albarrán-Ariagada F, Retamal J C, Solano E and Lamata L 2018 Measurement-based adaptation protocol with quantum reinforcement learning arXiv:1803.05340
[13] Melnikov A A, Nautrup H P, Krenn M, Dunjko V, Tiersch M, Zeilinger A and Briegel H J 2018 Active learning machine learns to create new quantum experiments Proc. Natl Acad. Sci. 115 6
[14] Settles B 2009 Active learning literature survey Computer Sciences Technical Report 1648 University of Wisconsin–Madison
[15] Wigley P B et al 2016 Fast machine-learning online optimization of ultra-cold-atom experiments Sci. Rep. 6 25890
[16] Ju S, Shiga T, Feng L, Hou Z, Tsuda K and Shimoi J 2017 Designing nanostructures for phonon transport via Bayesian optimization Phys. Rev. X 7 021024
[17] Li C, Leal D R d C, Rana S, Gupta S, Sutti A, Greenhill S, Slezak T, Height M and Venkatesh S 2017 Rapid Bayesian optimization for synthesis of short polymer fiber materials Sci. Rep. 7 5683
[18] Browaeys A, Barredo D and Lahaye T 2016 Experimental investigations of dipole–dipole interactions between a few Rydberg atoms J. Phys. B: At. Mol. Opt. Phys. 49 152001
[19] Nogrette F, Labuhn H, Ravets S, Barredo D, Béguin L, Vernier A, Lahaye T and Browaeys A 2014 Single-atom trapping in holographic 2D arrays of microtraps with arbitrary geometries Phys. Rev. X 4 021034
[20] Schlosser M, Tichelmans S, Kruse J and Birkl G 2011 Scalable architecture for quantum information processing with atoms in optical micro-structures Quantum Inf. Process. 10 907–24
[21] Wüster S and Rost J-M 2018 Rydberg aggregates J. Phys. B: At. Mol. Opt. Phys. 51 032001
[22] Weimer H, Müller M, Lesanovsky I, Zoller P and Büchler H P 2010 A Rydberg quantum simulator Nat. Phys. 6 382–8
[23] Labuhn H, Barredo D, Ravets S, de Léséleuc S, Macrì T, Lahaye T and Browaeys A 2016 Tunable two-dimensional arrays of single Rydberg atoms for realizing quantum Ising models Nature 534 667
[24] Barredo D, Labuhn H, Ravets S, Lahaye T, Browaeys A and Adams C S 2015 Coherent excitation transfer in a spin chain of three Rydberg atoms Phys. Rev. Lett. 114 113002
[25] Hague J P and MacCormick C 2012 Quantum simulation of electron–phonon interactions in strongly deformable materials New J. Phys. 14 033019
[26] Lesanovsky I 2012 Liquid ground state, gap, and excited states of a strongly correlated spin chain Phys. Rev. Lett. 108 105301
[27] Mülken O, Blumen A, Amthor T, Giese C, Reez-Lanour M and Weidemüller M 2007 Survival probabilities in coherent exciton transfer with trapping Phys. Rev. Lett. 99 090601
[28] Robicheaux F, Hernández J V, Topçu T and Noordam L D 2004 Simulation of coherent interactions between Rydberg atoms Phys. Rev. A 70 042703
[29] van Ditthuijen C S E, Koenderink A F, Hernández J V, Robicheaux F, Noordam L D and van den Heuvel H B v L 2008 Spatially resolved observation of dipole–dipole interaction between Rydberg atoms Phys. Rev. Lett. 100 243201
[30] Günter G, Schemp H, Robert-de Saint-Vincent M, Gavryusev V, Helmsrich S, Hofmann C S, Whitlock S and Weidemüller M 2013 Observing the dynamics of dipole-mediated energy transport by interaction-enhanced imaging Science 342 954
[31] Schönleber D W, Eisfeld A, Genkin M, Whitlock S and Wüster S 2015 Quantum simulation of energy transport with embedded Rydberg aggregates Phys. Rev. Lett. 114 123005
[32] Schemp H, Günter G, Wüster S, Weidemüller M and Whitlock S 2015 Correlated exciton transport in Rydberg-dressed-atom spin chains Phys. Rev. Lett. 115 093002
[33] Genkin M, Schönleber D W, Wüster S and Eisfeld A 2016 Non-Markovian dynamics in ultracold Rydberg aggregates J. Phys. B: At. Mol. Opt. Phys. 49 134001
[34] Schönleber D W, Bentley C D B and Eisfeld A 2018 Engineering thermal reservoirs for ultracold dipole–dipole-interacting Rydberg atoms New J. Phys. 20 013011
Beterov I I, Ryabtsev I I, Tretyakov D B and Entin V M 2009 Quasiclassical calculations of blackbody-radiation-induced depopulation rates and effective lifetimes of Rydberg nS, nP, and nD alkali-metal atoms with n< = 80 Phys. Rev. A 79 052504

Pedregosa F et al 2011 Scikit-learn: Machine learning in python J. Mach. Learn. Res. 12 2825–30

Helmrich S, Arias A, Pehoviak N and Whitlock S 2016 Two-body interactions and decay of three-level Rydberg-dressed atoms J. Phys. B: At. Mol. Opt. Phys. 49 03LT02

Schauss P 2014 High-resolution imaging of ordering in Rydberg many-body systems PhD Thesis LMU München: Fakultät für Physik