Machine-learning Online Optimisation for Evaporative Cooling in Cold-atom Experiments

R. Wang
(Dated: May 2, 2021)

As quantum systems become increasingly complex, optimisation algorithms are becoming a requirement for high-precision experiments. Machine-learning online optimisation offers an alternative to theoretical models, relying instead on experimental observations to continuously update an internal surrogate model. Two online optimisation techniques are reviewed in this paper in the context of evaporative cooling for the efficient and high-quality production of Bose-Einstein condensates (BEC). These two methods prioritise different stages of cooling with one focused on optimising experimental settings and the other on improving image acquisition.

I. INTRODUCTION

In 1925, Einstein predicted that a new quantum state of matter could condense out of a gas of integer spin particles (bosons) when cooled to temperatures close to absolute zero (nanokelvins) [1, 2]. Experimental observation of this process, called Bose-Einstein condensation, came 70 years later using an ultracold gas of rubidium atoms [3].

Despite being composed of atomic constituents, the quantum state of a Bose-Einstein condensate (BEC) can be collectively characterized by a single macroscopic wave function. Below a critical temperature, a large fraction of bosons occupy the lowest-energy state (ground state) and assume identical wave identities, acting like a 'super atom' that exhibits quantum behaviour at a macroscopic level. For this reason, degenerate gases are routinely used as quantum simulators to investigate a variety of quantum phenomena, such as many-body physics [4], non-equilibrium dynamics [5], phase transitions [6], superfluidity and superconductivity [7], and measurement sensitivity [8].

While methodologies for quantum gas experiments are well-established, the measurement rate is limited by long production times, typically lasting tens of seconds; the sampling process itself lasts for about a second before a destructive measurement is made [9]. Thus, producing BECs with short duty cycles is of particular interest and is essential for precise quantum sensors, such as atomic clocks and interferometers [10], pressure and inertial sensors [11, 12], and gravimeters [13].

II. MOTIVATION

The standard method to reach ultracold temperatures is by collisional evaporative cooling[1], which governs the duty cycle length. Evaporation requires the precise sequencing of time-varying magnetic and optical fields, parameterised by a time-specific value set for a given sequence. Although microscopic models exist to describe this process [14], such semi-classical theories can oversimplify dynamics, or miss non-intuitive yet more effective solutions [15] only discoverable through experimentation.

For most applications, data acquisition for high-precision experiments requires achieving the optimal measurement in a limited number of iterations. Given the large parameter space of a typical sequence, optimising experimental settings through exhaustive search is highly impractical. Instead, this discovery process is automated with *machine-learning online*[2] *optimisation* (MLOO). Isolating the atomic absorption signal from the noisy background is another prohibitive factor, which also benefits from the implementation of MLOO.

In this paper, we review two approaches to optimising cold-atom experiments: the first focusing on finding the optimal experimental settings, and the second on improving current absorption imaging techniques.

III. EVAPORATION MODEL

Evaporative cooling can be understood by analogy to cooling a cup of hot coffee by blowing on it. The speed of atoms in an atomic gas can be described by the Maxwell-Boltzmann distribution [16]. By removing the high-energy atoms[3], the remaining atoms re-equilibrate through elastic collisions, lowering the temperature of the sample. In hot coffee, the most energetic particles escape as vapour, taking with them their share of energy and thus temperature.

A. Runaway Evaporation

Efficient or *runaway* evaporative cooling requires the elastic collision rate γ_{el}, to be much larger than the loss rate

$$\gamma_{\text{el}} = \sigma\langle nv\rangle \gg \Gamma_{\text{loss}} \tag{1}$$

[1] Ultracold temperatures are usually achieved through a combination of laser/optical and forced evaporative cooling.

[2] *Online*, meaning optimisation that occurs in real-time.

[3] Atoms occupying the highest energy tail of the distribution.

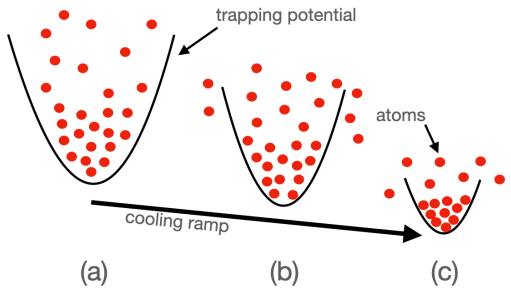

FIG. 1. The BEC evaporative cooling process of trapped atoms. In (a), high-energy atoms can move higher up the walls; (b) the walls are lowered and the most energetic atoms can spill over the walls and escape; (c) the remaining atoms rethermalize (collisionally). Steps (b) and (c) are repeated until the sample is sufficiently cold (determined by the critical temperature of a BEC). This diagram is based on https://cold-atoms.physics.lsa.umich.edu/projects/bec/evaporation.html.

where σ is the elastic scattering cross-section and $\langle n \rangle$ and $\langle v \rangle$ are the expectation values of the particle number density and velocity, respectively. As $\langle n \rangle \propto NT^{-\frac{3}{2}}$ and $\langle v \rangle \propto \sqrt{T}$ [17], the elastic collision rate thus varies as $\gamma_{\rm el} \propto NT^{-1}$ and is directly proportional to optical depth,

$$\tau = \ln\left(\frac{I_0}{I_t}\right) \tag{2}$$

by Beer's law[4]. The peak optical depth, $\tau_{\rm pk}$, for a cloud released and allowed to expand ballistically for a time t is described by

$$\tau_{\rm pk} = \frac{\lambda^2 m}{4\pi^2 k_B t^2}\frac{N}{T} \qquad [18] \tag{3}$$

Concerning optical imaging, a double-exposure scheme is generally used and two images are taken: the first with the cloud present, and the second reference exposure without. However, the noise patterns of the two images are not identical and thus results in residual noise in the final image [19]. Information on the optical depth (OD) along the line-of-sight is found through the difference in the logarithms of pixel counts (see Eqn. 2) between the two frames. Distinguishing the signal from the background becomes difficult, particularly for low-OD images. A novel single-exposure solution using a deep neural network (DNN) is presented in [19] (and discussed in §V).

B. BEC Observation

After switching off the trap, the condensate falls (gravity) and expands ballistically before an image is taken.

[4] I_0 is the incident tensity, and I_t is the transmitted intensity after time t.

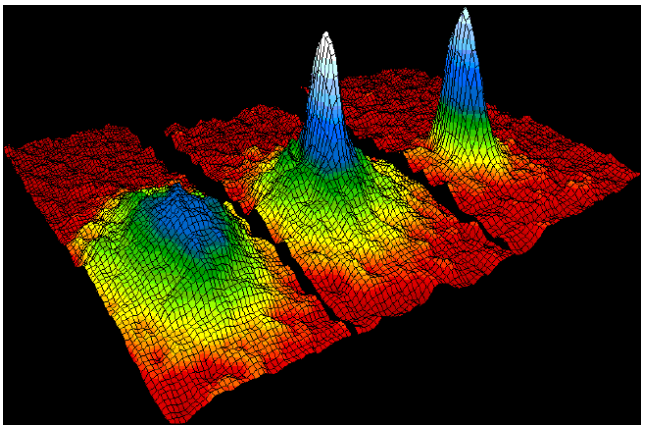

FIG. 2. 3-dimensional velocity distribution for a gas of rubidium atoms, showing successive snapshots in time, from the first confirmed 1995 production of a BEC by [3]. Atoms condense from less dense red/yellow/green areas to significantly denser blue/white areas. The central image is just after the appearance of a BEC; the left is before (non-condensed) and the right is a further evaporated and nearly pure condensate. Credit: NIST/JILA/CU-Boulder.

Following this 'time-of-flight' (TOF) expansion, the cloud is illuminated by a collimated resonant laser beam and the imaging shadow is recorded by a CCD camera. As the expansion dynamics of a quantum gas are distinctly different from those of a thermal gas, a bimodal density distribution is observed. By measuring the width of the cloud, the density profiles can be distinguished: thermal clouds have broad edges; as the sample cools and condenses into a dense atomic 'core', these edges become sharper [20, 21]. Observation of this characteristic bimodal distribution is evidence that a BEC has been produced (see Fig. 2).

In this case, we define an optimised quantum gas experiment as one that minimizes atomic loss while increasing the elastic collision rate to achieve runaway evaporation. For [21], this is achieved by maximising the atom number; for [20], this is evaluated by the sharpness of the cloud edges, with the cost bounded by optical depth (the lower and upper thresholds are determined by noise and saturation level, respectively). As the OD is directly proportional to the collision rate [17], absorption imaging has become the standard practise for characterising cold atomic gases. An absorption image provides the optical depth as a function of space. A simple inspection of the trend in peak optical depth in a few absorption images is enough to determine if evaporation is efficient [17], and thus if the BEC phase transition is reached.

$$\begin{cases} \uparrow \tau_{\rm pk},\ \uparrow \gamma_{\rm el} & \text{efficient, BEC reached} \\ \downarrow \tau_{\rm pk},\ \downarrow \gamma_{\rm el} & \text{inefficient, no BEC} \end{cases}$$

IV. MACHINE-LEARNING ONLINE OPTIMISATION (MLOO)

Let the parameter space be spanned by M experimental settings (e.g. voltage, laser parameters, timing, field strength [22]). A point in this space is represented by a vector $\mathbf{X} \in \mathbb{R}^M$. Each point has an associated cost $Y = f(\mathbf{X}) \in \mathbb{R}$, where minimising the cost function $f(\mathbf{X})$ guides optimisation toward the global optimum [21]. However, $f(\mathbf{X})$ is taken to be non-convex, thus it is possible that optimisation may converge to a local optimum. This can be rectified in part by increasing the number of optimisation cycles with varying initial conditions [21].

The experimental apparatus and optimisation loop begin with the trapped atomic cloud. The machine learner is given an initial vector $\mathbf{X}_0$ of experimental settings. The gas is transported into an ultra-high vacuum environment, where it is evaporatively cooled. Properties of the cloud (e.g. atom number [19] or width of cloud edges[5] [20]) are extracted from absorption images taken after TOF expansion and are used in evaluation of the cost. A new set of experimental parameters $\mathbf{X}_*$ is calculated based on the cost Y_0, to be used in the next sequence. Optimisation is terminated when there is no further improvement to the cost. Together, the experiment and learner form a closed loop. A diagram of this feedback loop can be found in Fig. 1 of [21] or [20].

While other optimisation techniques exist, these are often sub-optimal as most require accurate characterisation of the cloud (e.g. trap geometry, loss mechanisms) and/or apply over-simplifying assumptions (e.g. a highly truncated distribution[6], adiabaticity) [23] which may not necessarily hold for all instances. These procedures are often inflexible for special cases, such as dynamical traps [9] or the presence of dipolar interactions [24, 25]. Thus, most groups adopt a step-wise optimisation procedure, introducing incremental adjustments to parameters at each time step.

The following sections discuss various optimisation schemes, as examples of online optimisation (OO) in the context of BEC formation in cold-atom experiments. In order of appearance, the main papers referenced are [21], [20], and [19].

A. Differential Evolution

Inspired by biological evolution, differential evolutionary (DE) algorithms assess a population of candidate solutions based on their fitness. If M-dimensional vectors $\mathbf{X}_i$ (individuals) represent n sets of experimental settings – in the randomised set comprising the initial population, $\{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$ – the fitness of each settings vector is the experimentally-determined associated cost, Y_i. Random variations are introduced by *mutation*, and new vector candidates are generated by *crossover* (mixing) features of pre-existing individuals [21, 26].

In [21], a new, mutated vector appears as $\mathbf{V} = \mathbf{X}_k + (\mathbf{X}_i - \mathbf{X}_j)$, where vectors $\mathbf{X}_i$, $\mathbf{X}_j$, and $\mathbf{X}_k$ are randomly chosen. A new candidate vector $\mathbf{X}_*$ is produced by randomly picking elements from either $\mathbf{X}_i$ or $\mathbf{V}$. This crossover moves $\mathbf{X}_i$ to a new position in the search space, described by $\{\mathbf{X}_*, Y_*\}$. If $\mathbf{X}_i$ is an improved solution (i.e. $Y_* < Y_i$) then $\mathbf{X}_*$ replaces $\mathbf{X}_i$; else, it is discarded. The process repeats until a global minimum is found.

B. Gaussian Process Regression

Bayesian optimisation uses statistical models to predict optimal parameters, where decisions are made with all previous evaluations of $f(\mathbf{X})$ taken into account. The approach is to build an internal surrogate model for $f(\mathbf{X})$ at each instance, which informs the learner's decision on the next point in $\mathbf{X}$ to evaluate $f(\mathbf{X})$. This is especially pertinent when $f(\mathbf{X})$ is expensive to evaluate, as every experimental observation is used to improve the model and is not solely dependent on derivative information (e.g. local gradient (first-order) and Hessian approximations (second-order)) [27]. This method is used by both [20] and [21], the results of which are both discussed in § IV D.

The most common and well-studied[7] class of surrogate models are Gaussian Process (GP) models. These models are favoured for their strong generalisability, tractability, and flexible non-parametric inference [32], making them suitable for treating complex regression problems such as small samples and non-linearities [33]. A GP infers a probability distribution in function space, rather than over individual (function) parameters. Based on new data, GP regression uses Bayes' rule to update the hypothesised prior distribution. To choose the next point of interest (POI), a predictive posterior distribution can be computed from both the prior and dataset.

1. Covariance Function

A stochastic process with the property that any finite collection of variables (or equivalently, any linear combination) $[f(\mathbf{X}_1), \ldots, f(\mathbf{X}_N)]$, is normally distributed is

[5] [20] argues that atom number and temperature are inadequate measures, as accurately determining these quantities near condensation becomes challenging with very few runs per parameter set. Instead, the width and sharpness of the edges of the cloud are measured from the optical depth as a function of space.

[6] The truncation parameter, η, assumes atoms with $E > \eta k_B T$ evaporate instantly.

[7] The use of GP priors is well-established, dating back to the '60-'70s [28–30]. As such, only a brief review is provided here. For a more thorough introduction, the reader is directed to [31].

referred to as a Gaussian Process [34]. Properties of a GP $f : \mathcal{X} \to \mathbb{R}$, where $\mathcal{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_N)$ and is a subset of $\mathbb{R}^N$, are determined by a mean function $M : \mathcal{X} \to \mathbb{R}$ and a positive definite kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that defines the covariance [27, 34].

The default for K is often the Gaussian (squared exponential) kernel

$$K(\mathbf{X}_i, \mathbf{X}_j) = \exp\left\{-\frac{1}{2}\sum_{k=1}^{M}\left(\mathbf{X}_i[k] - \mathbf{X}_j[k]\right)^2 / {h_k}^2\right\} \quad (4)$$

where $\mathbf{X}_i[k]$ is the k^{th} element in the vector $\mathbf{X}_i$ and h_k belongs to a set $H = (h_1, \ldots, h_M)$ of correlation lengths, the hyperparameters to be fitted online (see § IV D 1 for experimental results).

In optimising experimental settings, GP regression is used to fit the function that maps these settings to the empirical cost. For [21], the system is initialized with a training set (generated by DE) of $2M$ settings, in the form of cost pairs $\{\mathbf{X}_i, Y_i\}$. By mapping, the estimated cost (and uncertainty) of any $\mathbf{X}_*$ can be found from the GP fit; exploration into new settings is steered by the lowest predicted cost. A comparison of h_k values across all settings can be made by normalising each $\mathbf{X}[k]$ with respect to the extremal (min/max) values of the k^{th} setting [21].

2. Acquisition Functions

In general, Bayesian acquisition functions depend on all previous observations and the GP hyperparameters to guide the search for the optimum [27, 31]. The only dependence on the model is through its predictive mean and variance functions. There are several optimisation strategies:

1. *Probability of improvement* – An intuitive approach suggested by [35] is to maximise the probability of improvement over the current best value.

2. *Expected improvement* (EI) – A similar strategy is to maximise the expected improvement over the current best [27].

3. *GP upper confidence bound* (GP-UCB) – Alternatively, an acquisition function can be chosen such that it balances: (i) improving the model (*exploration*) and (ii) using the model to find the global optimum (*exploitation*).

   Points may be selected on the basis of maximising the UCB [31]:

   $$\text{UCB}(\mathbf{X}) = \mu(\mathbf{X}) + \kappa\sigma(\mathbf{X}) \quad (5)$$

   where κ may be tuned to balance exploration versus exploitation. The learner explores actions with high uncertainty and exploits actions with the highest reward. To optimise the evaporative cooling of thulium atoms, [23] employed this method to achieve BEC efficiently.

Other choices of acquisition functions exist, such as the instantaneous regret function [36], knowledge-gradient [37, 38], or (predictive [39]) entropy search [40], etc., but are not mentioned here.

C. Artificial Neural Network

As a black-box function approximator, an Artificial Neural Network (ANN) provides a mapping between an input, – in this case, X settings vectors – and an output, the associated costs Y. In [21], the activation function for each node was selected to be the Gaussian Error Linear Unit (GELU). A suitable choice of the structure and scale of the ANN should consider the complexity and size of the vector inputs, while maintaining computational efficiency. The ANN utilised by [21] consisted of 3 hidden layers of 8 fully-connected neurons. To update the ANN, the *Adam* algorithm for stochastic optimisation was chosen. Again, the system is trained with $2M$ settings generated by DE, and iterates through a maximum of 35 sequences.

1. Gaussian Error Linear Units

The Gaussian Error Linear Unit (GELU) proposed by [41] merges the functionalities of dropout, zoneout [42], and ReLU's such that the transformation applied to the neuron input x is stochastic yet also dependent on the input. That is, the GELU nonlinearity weighs by value, with inputs having a higher probability of being dropped with decreasing x. The nonlinearity arises from the *deterministic* counterpart of a *stochastic* regulariser: the expected transformation of a stochastic regulariser on an input x governs the scaling. The activation function then takes the form [41]

$$\Phi(x) \times Ix + (1 - \Phi(x)) \times 0x = x \cdot \Phi(x) \quad (6)$$

where $\Phi(x)$ is the cumulative distribution function (CDF) of the standard Gaussian distribution. The GELU is defined as [41]

$$\text{GELU}(x) = xP(X \leq x) = \frac{x}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \quad (7)$$

which may be approximated as

$$\approx 0.5x\left[1 + \tanh\sqrt{2/\pi}\left(x + 0.044715x^3\right)\right] \quad (8)$$

or

$$\approx x \cdot \sigma(x) \qquad \text{where } \sigma(x) = 1/(1 + e^{-x}) \quad (9)$$

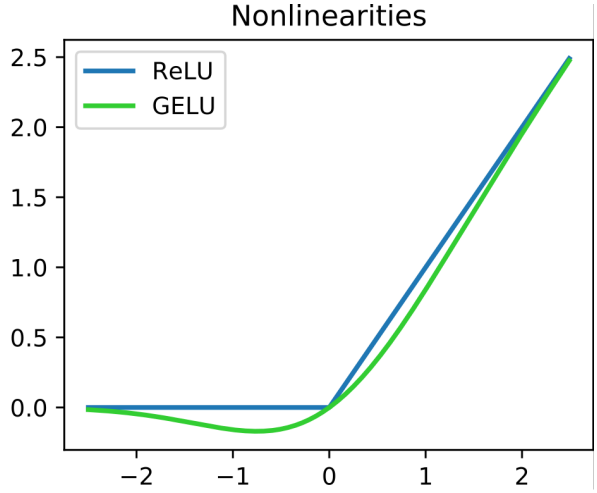

FIG. 3. Plot of ReLU and GELU near $x = 0$. ([43], CC BY-SA 4.0)

if speed is favoured over accuracy. Other CDFs may be used, for example: the standard logistic CDF $\sigma(x)$, which gives the SiLU $x \cdot \sigma(x)$ (Sigmoid Linear Unit) or alternatively, the normal distribution with tunable μ and σ^2 hyperparameters.

While GELU is similar to ReLU and ELU to some degree, [41] found that GELU matched or outperformed both in completing a variety of tasks (CV, NLP, ASR). A plot of both the GELU and ReLU nonlinearities is shown in Fig. 3. In implementing GELU, [41] offers two practical tips:

1. Use an optimiser with momentum, as is the standard practise for training DNN.

2. When using a different CDF, the approximation must be close to the CDF of a Gaussian distribution. If the closeness is insufficient, performance will be negatively impacted.

The code repository is provided by D. Hendrycks (while at TTIC) and is available at https://github.com/hendrycks/GELUs.

2. *Adam Optimisation*

Adam (adaptive moment estimation) [44] merges the advantages of two popular optimisation methods: (i) AdaGrad [45], which handles sparse gradients and (ii) RMSProp [46], which deals with non-stationary objectives and excels in online settings. The result is a computationally efficient and effective algorithm for gradient-based optimisation of noisy cost functions. Adam is well suited for solving problems with large amounts of data and/or parameters, and a wide range of non-convex problems quickly with comparatively fewer resources than other methods. As adaptive methods based on exponential moving averages (EMA), like RMSProp or Adam, are very popular methods for training deep neural networks, the reader is directed to [44], [32], and [47] for review.

D. MLOO Performance

Documentation for M-LOOP, an open-source optimisation package, is available at https://m-loop.readthedocs.io/en/stable/ and code at https://github.com/michaelhush/M-LOOP. In addition, Supplementary Information for this paper is also provided and contains equations for GP process evaluation and experimentally determined optimal values for a set of 16 parameters.

1. *Comparison to the Nelder-Mead Optimiser*

The performance of MLOO using a GP process statistical model by [20] is compared with a method used previously in optimising gate fidelity [48], the Nelder-Mead (NM) solver [49]. Predictions for the mean function and variance are fit by the sequential Monte Carlo[8] (SMC) method of particle learning (PL). A weighted average is performed over $\mathcal{H} = \{H_1, \ldots, H_p\}$, the 'hypothesis set' of hyperparameters where each hypothesis is treated as a particle [20]. The weighted functions used by [20] are defined as follows:

$$M_{\hat{C}}(\mathcal{X}|\mathcal{O},\mathcal{H}) \equiv \sum_{i=1}^{P} w_i \mu_{\hat{C}}(\mathcal{X}|\mathcal{O},\mathcal{H}_i) \tag{10}$$

where $M_{\hat{C}}$ is the weighted mean function, $\mathcal{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_N)$, $\mathcal{C} = (C_p, \ldots, C_N)$, and $\mathcal{U} = (U_1, \ldots, U_N)$, which comprise the observation set $\mathcal{O} = (\mathcal{X}, \mathcal{C}, \mathcal{U})$ of the sets of all parameters, costs, and uncertainty. The $\hat{}$ denotes the weighted counterparts of each function.

The weighted variance function is

$$\Sigma^2_{\hat{C}}(\mathbf{X}|\mathcal{O},\mathcal{H}) \equiv \sum_{i=1}^{P} w_i \left[\sigma^2_{\hat{C}}(\mathbf{X}|\mathcal{O},\mathcal{H}_i) + {\mu_{\hat{C}}}^2(\mathbf{X}|\mathcal{O},\mathcal{H}_i)\right] - M^2_{\hat{C}}(\mathbf{X}|\mathcal{O},\mathcal{H}_i) \tag{11}$$

with relative weights w_i.

2. *Optimisation Strategies*

The learner has two choices:

[8] Although Markov chain Monte Carlo (MCMC) is typically chosen for optimisation problems, it is unfit for OO given its memoryless property (although many methods exist to accelerate MCMC for online implementation [50–53]).

1. *Minimise* $M_{\hat{C}}(\mathbf{X})$ and prioritise optimisation, but may not converge to the global optimum – an 'optimiser'.

2. *Maximise* $\Sigma^2_{\hat{C}}$ and investigate areas in which learner is most uncertain, formulating hypotheses and updating the model based on experimental data – a 'scientist'.

or alternatively, a blended choice

3. *Minimise* $B_{\hat{C}}(\mathbf{X}) \equiv bM_{\hat{C}}(\mathbf{X}) - (1-b)\Sigma^2_{\hat{C}}$

where b steps linearly from $b = 0$ and $b = 1$, the 'optimiser' and 'scientist' strategies, over one sequence. If the change between the previous and updated sets of experimental settings is too drastic, no atoms are produced in virtually all experiments thereafter. Thus, the learning rate is restricted but the exploration space remains unbounded [20].

Two parameterisations of the evaporation ramp are used by [20]:

(i) simple (linear) control over the amplitudes of the start and end points of the ramp

$$\mathscr{R}_s(y_i, y_f, t_f) = y_i + (y_f - y_i)\frac{t}{t_f} \tag{12}$$

(ii) complex (polynomial), an extension of the simple case with polynomial terms of degree $d \geq 3, 4, 5$

$$\begin{aligned}\mathscr{R}_c(y_i, y_f, A_2, A_3, A_4, t_f) = y_i + (y_f - y_i)\frac{t}{t_f} \\ + A_2(t-t_f) + A_3(t-t_f)\left(t + \frac{1}{2}t_f\right) \\ + A_4 t(t-t_f)\left(t + \frac{2}{3}t_f\right)\left(t + \frac{1}{3}t_f\right)\end{aligned} \tag{13}$$

where A_2, A_3, A_4 are relabelled by [20] to match the quadratic, cubic, and quartic terms; these variables are also referred to as A_1, A_2, A_3 respectively by [20], which is the convention we will use from this point forward (see Supplementary Information).

For all three ramps, [20] used complex parameterisation (Eqn. 13), and also included an additional parameter t_f which marks the final time of the cooling ramp. While the parameters y_i, y_f, A_1, A_2, A_3 of each are independent, t_f is common between all ramps. This gives a total of 16 parameters, the optimum values of which may be found in Table 1 of Supplementary Information. A comparison between the brute force, Nelder-Mead, and MLOO methods for a set of 16 parameters is given in Table I.

Compared to the NM solver (Table I), the learner discovered BEC ramps in only a few experimental runs. This was achieved by (a) using only the best hypothesis set ($P = 1$) to update the model and (b) prioritise fitting H for each of the 3 most important parameters

	Parameters	Runs
Brute force	16	10^{16}
Nelder-Mead	16	145
MLOO	16	10

TABLE I. Experimental results from the first implementation of MLOO by creators [20] in ultra-cold atom experiments. Both the NM and MLOO optimisers were trained for 20 runs using a common set of parameters.

(end points of ramps). However, a drawback is the poor fitting of the other correlation lengths, leading to uninformed estimates and unreliable predictions.

Another trade-off is to improve estimations of correlation lengths by increasing the particle number (to $P = 16$) [54], but also seeing an increase run time. This can be compensated for by using the simple parameterisation of ramps and about half as many parameters (a total of 7). In obtaining a more reliable estimate, the convergence rate is slowed. However, it is still faster than the NM optimiser (see Figure 2 of [20]). The least sensitive of the 7 parameters does not influence BEC production; it was identified by the learner and removed from the experimental design. With 6 parameters, the learner performs better than the 7 parameter case, converging faster and producing a higher quality BEC [20]. From the results of [20], lower parameter searches converged to similar solutions, while higher dimensional searches led to noticeably different optima. A simple summary of the three optimisation runs (out of a total of 5) can be found in Table II.

Particles	Parameters	Optimisation Strategy
1	16	Complex
16	7	Simple
16	6	Simple

TABLE II. A summary of values for each of the three optimisation runs, used by [20] and discussed in §IV D 1.

3. *Convergence Rates to BEC*

While [20] prioritised maximising BEC production and quality, [21] favours a faster convergence rate and sets a threshold atom number for producing a BEC. The chosen cost function takes the form

$$f(\tilde{N}) = -\frac{\left(1 + \arctan(\tilde{N} - \tilde{N}_0)\right)}{1 + \tilde{t}} \tag{14}$$

where $\tilde{N}_0$ is the threshold atom number, chosen to be a BEC of size 1×10^5 (comparable to a BEC achieved by manual tuning; see Table I), and $\tilde{t}$ is the sequence duration [21]. This cost function is tailored for optimising convergence rates: it rewards short sequence times $\tilde{t}$, gives little reward to a BEC with an atom number $> \tilde{N}_0$,

and penalises a BEC that does not reach $\tilde{N}_0$. With OO, [21] found that optimal settings based on Eqn. 14 produced a BEC of 9.6×10^4 atoms and reduced sequence time by 20% (from 58s to 46s).

Instead of NM, a baseline for comparison is established by choosing randomised initial settings (and thus does not produce an atomic cloud) for each learner. For each method, one optimisation routine is performed until no further improvement is found within 35 cycles or a time limit of 3 hours. Experimental results from [21] are presented in Table III.

	Runs	Atom Number
Manual	–	1.1×10^5
DE	DNC	–
GP	47	3.8×10^5
ANN	117	3.2×10^5

TABLE III. Comparison of convergence rates between methods. DE did not converge (DNC) within the time limit of roughly 3 hours (or a maximum of 180 sequences). The convergence rate of ANN is between those of GP and DE. For both GP and DE, the quoted number omits the DE-generated training set of $2M = 70$ sequences.

When the cost function drops below roughly 9.2, a bimodal density distribution is observed in the cloud – a signature of BEC. The experimental convergence rates of each method is presented are Table IV. The relative convergence rates appear as:

- GP (fastest): while it is the most rapidly converging of the 3 methods tested, the number of sequences (and thus time) increases with the number of parameters; fitting multiple GPs is computationally expensive.

- ANN (slower): the relative slowness of ANN compared to the GP method can be attributed to the large datasets needed to train a fully-connected network.

- DE (slowest): the simplest method that incorporates an element of randomness when choosing the next POI. Thus, the chance that the optimiser begins with good settings early on should be taken into account (and is, by adjusting the minimum BEC costs. See Table IV).

	Runs	Minimum Cost
DE	156	9.2
GP	14	9.4
ANN	75	9.6

TABLE IV. A comparison of the number of sequences needed to achieve the BEC cost threshold of approximately 9.2 (though varies slightly for the GP and ANN methods).

V. SINGLE-SHOT ABSORPTION IMAGING

In contrast with the MLOO methods discussed previously, which prioritised optimising experimental parameters, a novel approach using deep learning is proposed by [19] to optimise image acquisition. The typical procedure is to obtain two successive absorption images of the system, one with the atomic cloud present and one without (as described in §III A). A complication of this double-exposure approach is the presence of structured residual noise patterns, such as Newton's rings, in the final image. A unique single-shot solution offered (and demonstrated[9]) recently (2020) by [19] employs a deep neural network (DNN) to improve the signal quality. Although the imaging sequence performed by [19] does not include OO, it is easy to extrapolate their methodology to a continuously updated model.

A. Extracting Observables

The experimental procedure for cooling in [19] is the same as those used by [55] (to optimise optical transfer of atoms) and [56] (RF spectroscopy sensitivity). Images acquired and used in both the conventional and single-shot techniques are presented in Table V.

Frame Types	Use in Experiment	Imaging Procedure
Without atoms	Training ("ground truth" values) and validation of the NN	DNN
1^{st} (raw) exposure	To be used in DNN image reconstruction (input and prediction)	DNN/ Conventional
2^{nd} exposure	To compare the single-shot and double-shot techniques (reference)	Conventional
Dark	Zero reference (no illumination)	DNN

TABLE V. List of acquired images and the associated applied imaging technique. In [19], the first image was illuminated by an 80 μs pulse. The second (reference) exposure was taken 50 ms later, after the atoms have moved out of the CCD field of view. Note that the 2^{nd} exposure is only needed for comparison to the conventional method and is in no way used in the DNN technique. Images were recorded with a 14-bit CCD camera.

The two observables of interest, the atom number N and temperature T, are controlled by the final trap depth of the ramp and extracted from the momentum distribution. The distinct difference in expansion dynamics between a quantum versus thermal gas is seen in the resulting BEC signature (bimodal) density distribution

[9] [19] performed their single-shot imaging scheme on an ultracold, quantum degenerate Fermi gas of ^{40}K and found noticeable improvements in the SNR.

(see §III B) after TOF expansion. For [19], the OD images are fitted with [57]

$$\tau(x,y) = \tau_{\rm pk} \cdot \frac{\mathrm{Li}_2\left(-z\exp -\frac{(x-x_0)^2}{2\sigma_x^2} - \frac{(y-y_0)^2}{2\sigma_y^2}\right)}{\mathrm{Li}_2(-z)} + B \quad (15)$$

where $\mathrm{Li}_n(z)$ is the polylogarithm (Jonquière's function), B is the residual background, and $z = e^{\mu/k_B T}$ is the fugacity, from which the T is obtained. Integrating over the fitted momentum distribution gives N [19]. This can be seen in the physical interpretation of the cost function from [21], where the number of atoms with momentum close to zero increases as BEC production improves.

B. DNN Architecture, Training, and Optimisation

A summary of the image transformation process and DNN pipeline is provided in Fig. 4. From the masked OD image (input), a DNN prediction is made through transformed and transposed convolutions. Since recovering the spatial density profile from the masked region is of interest, the goal of the U-Net [58] convolutional network is to optimise noise-pattern reconstruction. As an unsupervised learner, the baseline or "ground truth" is established by using images without atoms. Reconstruction is achieved by minimising differences between the ground truth and the network's prediction or equivalently, by minimising the mean squared error (RMSE) loss function. By comparing predictions to the ground truth values at each step and adjusting the weights accordingly, an optimised model is produced [19].

Predictions on images with atoms can be inferred by the optimised model, even on few data sets and effectively no prior knowledge of the system (except for knowledge of the absence of atoms in the periphery). In training, the network has additional knowledge on the atoms in the masked region from the ground truth values. The masked region is 190 pixels in diameter, and a factor of 2 larger than that of a typical cloud; this is to ensure that the peripheral area used in DNN background prediction is completely devoid of any absorption signal [19]. ADAM [44] and Glorot [59] were used for parameter optimisation. Values used in the network are provided in Table VI.

Parameters	**Layers**	**Frames**	
		No atoms	**Validation and RMSE**
20×10^6	27	30×10^3	7×10^3

TABLE VI. A description of the experimental framework of [19].

C. Performance

The performance of the DNN is evaluated based on residual noise, and compared with other methods, the double-exposure and primary component analysis (PCA) techniques.

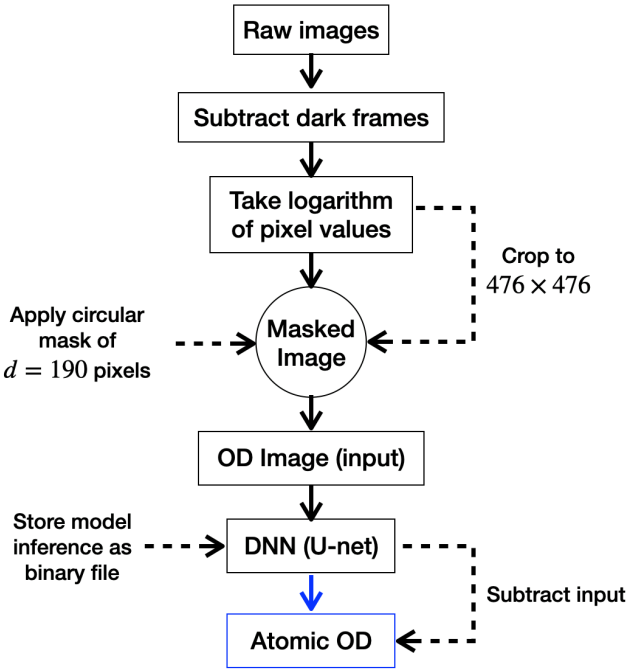

FIG. 4. Architecture of image transformation and the DNN pipeline (a summary of the experimental procedure performed by [19]).

1. *Validation Set and Other Techniques*

The decay of residual error between the DNN prediction and ground truth is analysed as a function of the number of training iterations (epochs). After a few hundred epochs, the initial RMSE decay rate is noticeably slowed and thus training is truncated at 1133 epochs [19]. The relative differences in residual loss between the DNN-based, conventional, and PCA techniques can be seen in the peaks of the residual error distribution (see Figure 3 in [19]). The DNN-based single-shot method supersedes the conventional method on the validation set, showing a comparatively lower RMSE and therefore better performance.

2. *Degenerate Fermi Gas*

In imaging a degenerate Fermi gas of ^{40}K, the DNN technique was able to remove residual fringes from the final image, whereas the double-shot technique did not (see Figure 5 of [19] for comparison images). This was tested for varying trap depths with a cloud of 30×10^3 atoms (and also for a variable number of atoms), where [19] showed that the single-exposure approach still outperformed the conventional double-exposure scheme.

In extracting physical observables, namely atom number and temperature, the DNN method did not introduce any new sources of systematic error. In fact, the uncertainty in extracting both N and T was found to be smaller by $\sim 17\%$ using the new method. It should be noted that this improvement in RMSE extract error is compared using an average error over 10 experimental runs using 5 different trap depths [19]. A method based on Bayesian inference has been proposed by [34] for quantum systems with poor statistics (and where the Gaussian noise assumption is inappropriate), even in the limit of single-shot measurements like absorption imaging.

The open-source Python software package and MATLAB script can be found at https://absdl.github.io/ and is easily implemented on any imaging apparatus (after training).

VI. DISCUSSION AND FUTURE OUTLOOK

Many optimisation strategies exist for optimising ultra-cold atom experiments with quantum degenerate gases. The focus of the optimisation, however, may vary from case to case. For example, the methods described in §IV prioritise fast convergence rates and thus search for optimal experimental settings. The motivation for achieving BEC in only a few runs may differ as well, with some aiming to minimise temperature, while others evaluate based on the atom number [21] or the width of cloud edges [20]. Different stages of the cooling sequence may be examined as well, such as optimising data acquisition [20, 21, 23] or signal processing [19]. These often require implementing different procedures, using MLOO or DNNs to update a predictive model of the optimal system.

For all of these optimisation techniques, online optimisation can offer improvements in many different respects. As an example and potential extension, the model from the DNN-based single-shot method may be continuously updated as new images arrive with online optimisation. When compared with other machine learning algorithms – Nelder-Mead [20], differential evolution, GP regression and ANNs – those employing MLOO consistently achieved better results (e.g. in BEC quality and production speed) than previously established approaches. The advantage of MLOO largely comes from building an internal model for inference, which dramatically decreases system characterisation and analysis overhead in optimisation.

An extension to the use of MLOO can be found in almost any high-precision experiment, where having precise control over quantum systems is imperative. Optimisation of quantum control has traditionally relied on theoretical modeling. However, with the growing complexity of quantum systems, it becomes more and more difficult or unrealistic to produce an accurate theoretical model. As such, the relevance and usefulness of optimised quantum control that is updated by experimental data have become increasingly apparent.

[1] Albert Einstein. Quantentheorie des einatomigen idealen gases. Sitzungsberichte der Preussischen Akademie der Wissenschaften, 1:3, July 1925.

[2] Satyendra Nath Bose. Plancks gesetz und lichtquantenhypothese. 1924.

[3] Mike H Anderson, Jason R Ensher, Michael R Matthews, Carl E Wieman, and Eric A Cornell. Observation of bose-einstein condensation in a dilute atomic vapor. science, 269(5221):198–201, 1995.

[4] Immanuel Bloch, Jean Dalibard, and Wilhelm Zwerger. Many-body physics with ultracold gases. Reviews of modern physics, 80(3):885, 2008.

[5] Tim Langen, Remi Geiger, and Jörg Schmiedmayer. Ultracold atoms out of equilibrium. Annu. Rev. Condens. Matter Phys., 6(1):201–217, 2015.

[6] Immanuel Bloch, Jean Dalibard, and Sylvain Nascimbene. Quantum simulations with ultracold quantum gases. Nature Physics, 8(4):267–276, 2012.

[7] Henrik Olofsson, Sven Åberg, and Patricio Leboeuf. Semiclassical theory of bardeen-cooper-schrieffer pairing-gap fluctuations. Physical review letters, 100(3):037005, 2008.

[8] Pinrui Shen, Kirk William Madison, and James Booth. Refining the cold atom pressure standard. Metrologia, 2021.

[9] Richard Roy, Alaina Green, Ryan Bowler, and Subhadeep Gupta. Rapid cooling to quantum degeneracy in dynamically shaped atom traps. Physical Review A, 93(4):043403, 2016.

[10] GM Tino, L Cacciapuoti, K Bongs, Ch J Bordé, P Bouyer, H Dittus, W Ertmer, A Gorlitz, M Inguscio, A Landragin, et al. Atom interferometers and optical atomic clocks: New quantum sensors for fundamental physics experiments in space. Nuclear Physics B (Proceedings Supplements), 166:159–165, 2007.

[11] Pinrui Shen, Kirk W Madison, and James L Booth. Realization of a universal quantum pressure standard. Metrologia, 57(2):025015, 2020.

[12] Jacob Dunningham, Keith Burnett, and William D Phillips. Bose–einstein condensates and precision measurements. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 363(1834):2165–2175, 2005.

[13] JE Debs, PA Altin, TH Barter, Daniel Doering, GR Dennis, Gordon McDonald, RP Anderson, JD Close, and NP Robins. Cold-atom gravimetry with a bose-einstein condensate. Physical Review A, 84(3):033610, 2011.

[14] CA Sackett, CC Bradley, and RG Hulet. Optimization of evaporative cooling. Physical Review A, 55(5):3797, 1997.

[15] Takahiko Shobu, Hironobu Yamaoka, Hiromitsu Imai, Atsuo Morinaga, and Makoto Yamashita. Optimized evaporative cooling for sodium bose-einstein condensation against three-body loss. Physical Review A, 84(3):033626, 2011.

[16] SJ Blundell and KM Blundell. Concepts in thermal physics (oup), 2010.

[17] JE Lye, CS Fletcher, U Kallmann, HA Bachor, and

JD Close. Images of evaporative cooling to bose-einstein condensation. Journal of Optics B: Quantum and Semiclassical Optics, 4(1):57, 2002.

[18] P Hannaford and RJ McLean. Atomic absorption with ultracold atoms. Spectrochimica Acta Part B: Atomic Spectroscopy, 54(14):2183–2194, 1999.

[19] Gal Ness, Anastasiya Vainbaum, Constantine Shkedrov, Yanay Florshaim, and Yoav Sagi. Single-exposure absorption imaging of ultracold atoms using deep learning. Physical Review Applied, 14(1):014011, 2020.

[20] Paul B Wigley, Patrick J Everitt, Anton van den Hengel, John W Bastian, Mahasen A Sooriyabandara, Gordon D McDonald, Kyle S Hardman, Ciaron D Quinlivan, P Manju, Carlos CN Kuhn, et al. Fast machine-learning online optimization of ultra-cold-atom experiments. Scientific reports, 6(1):1–6, 2016.

[21] Adam J Barker, Harry Style, Kathrin Luksch, Shinichi Sunami, David Garrick, Felix Hill, Christopher J Foot, and Elliot Bentine. Applying machine learning optimization methods to the production of a quantum gas. Machine Learning: Science and Technology, 1(1):015007, 2020.

[22] Yue Shen. Determination of the atom's excited-state fraction in a magneto-optical trap. PhD thesis, University of British Columbia, 2018.

[23] ET Davletov, VV Tsyganok, VA Khlebnikov, DA Pershin, DV Shaykin, and AV Akimov. Machine learning for achieving bose-einstein condensation of thulium atoms. Physical Review A, 102(1):011302, 2020.

[24] Antoine Browaeys, Daniel Barredo, and Thierry Lahaye. Experimental investigations of the dipolar interactions between single rydberg atoms. arXiv preprint arXiv:1603.04603, 2016.

[25] L Caldwell and MR Tarbutt. Enhancing dipolar interactions between molecules using state-dependent optical tweezer traps. Physical Review Letters, 125(24):243201, 2020.

[26] Pradnya A Vikhar. Evolutionary algorithms: A critical review and its future prospects. In 2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC), pages 261–265. IEEE, 2016.

[27] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. arXiv preprint arXiv:1206.2944, 2012.

[28] Theophilos Cacoullos. Estimation of a multivariate density. Technical report, University of Minnesota, 1964.

[29] Anthony O'Hagan. Curve fitting and optimal design for prediction. Journal of the Royal Statistical Society: Series B (Methodological), 40(1):1–24, 1978.

[30] Jose M Bernardo. Reference posterior distributions for bayesian inference. Journal of the Royal Statistical Society: Series B (Methodological), 41(2):113–128, 1979.

[31] Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arXiv preprint arXiv:1012.2599, 2010.

[32] Shiliang Sun, Zehui Cao, Han Zhu, and Jing Zhao. A survey of optimization methods from a machine learning perspective. IEEE transactions on cybernetics, 50(8):3668–3681, 2019.

[33] Christopher KI Williams. Gaussian processes for machine learning. Taylor & Francis Group, 2006.

[34] Frederic Sauvage and Florian Mintert. Optimal quantum control with poor statistics. PRX Quantum, 1(2):020322, 2020.

[35] Harold J Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. 1964.

[36] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. arXiv preprint arXiv:0912.3995, 2009.

[37] Peter I Frazier, Warren B Powell, and Savas Dayanik. A knowledge-gradient policy for sequential information collection. SIAM Journal on Control and Optimization, 47(5):2410–2439, 2008.

[38] Warren Scott, Peter Frazier, and Warren Powell. The correlated knowledge gradient for simulation optimization of continuous parameters using gaussian process regression. SIAM Journal on Optimization, 21(3):996–1026, 2011.

[39] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. arXiv preprint arXiv:1406.2541, 2014.

[40] Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. Journal of Machine Learning Research, 13(6), 2012.

[41] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2020.

[42] David Krueger, Tegan Maharaj, János Kramár, Mohammad Pezeshki, Nicolas Ballas, Nan Rosemary Ke, Anirudh Goyal, Yoshua Bengio, Aaron Courville, and Chris Pal. Zoneout: Regularizing rnns by randomly preserving hidden activations, 2017.

[43] Wikimedia Commons. Plot of the relu rectifier (blue) and gelu (green) functions near x = 0, 2020. File: ReLU_and_GELU.svg.

[44] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[45] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. Journal of machine learning research, 12(7), 2011.

[46] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent.

[47] S Reddi, Manzil Zaheer, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. In Proceeding of 32nd Conference on Neural Information Processing Systems (NIPS 2018), 2018.

[48] Julian Kelly, R Barends, B Campbell, Y Chen, Z Chen, B Chiaro, A Dunsworth, Austin G Fowler, I-C Hoi, E Jeffrey, et al. Optimal quantum control using randomized benchmarking. Physical review letters, 112(24):240504, 2014.

[49] John A Nelder and Roger Mead. A simplex method for function minimization. The computer journal, 7(4):308–313, 1965.

[50] Carlo Berzuini, Nicola G Best, Walter R Gilks, and Cristiana Larizza. Dynamic conditional independence models and markov chain monte carlo methods. Journal of the American Statistical Association, 92(440):1403–1412, 1997.

[51] Christian P Robert, Víctor Elvira, Nick Tawn, and Changye Wu. Accelerating mcmc algorithms. Wiley Interdisciplinary Reviews: Computational Statistics, 10(5):e1435, 2018.

[52] Thejs Brinckmann and Julien Lesgourgues. Montepython 3: boosted mcmc sampler and other features. Physics of the Dark Universe, 24:100260, 2019.

[53] Robert Salomone, Matias Quiroz, Robert Kohn, Mattias

Villani, and Minh-Ngoc Tran. Spectral subsampling mcmc for stationary time series. In International Conference on Machine Learning, pages 8449–8458. PMLR, 2020.

[54] Robert B Gramacy and Nicholas G Polson. Particle learning of gaussian process models for sequential design and optimization. Journal of Computational and Graphical Statistics, 20(1):102–118, 2011.

[55] Gal Ness, Constantine Shkedrov, Yanay Florshaim, and Yoav Sagi. Realistic shortcuts to adiabaticity in optical transfer. New Journal of Physics, 20(9):095002, 2018.

[56] Constantine Shkedrov, Yanay Florshaim, Gal Ness, Andrey Gandman, and Yoav Sagi. High-sensitivity rf spectroscopy of a strongly interacting fermi gas. Physical review letters, 121(9):093402, 2018.

[57] Wolfgang Ketterle and NJ Van Druten. Advances in atomic, molecular, and optical physics. 1996.

[58] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[59] XAYG Glorot, Y Bengio, YW Teh, and M Titterington. Proceedings of the thirteenth international conference on artificial intelligence and statistics. PMLR, 9:249–256, 2010.