Introduction

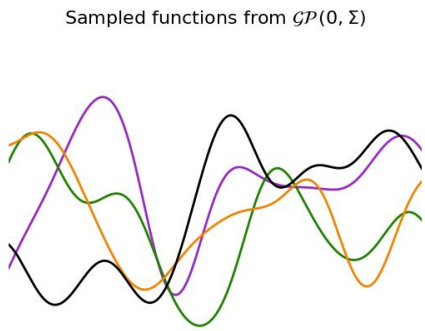
This project concerns itself with the problem of global optimization of expensive back-box functions in vast configuration spaces. The combination of these two properties, lead to a problem, that is exceedingly difficult to solve. The expensiveness of the black-box functions, means that it is not feasible to evaluate it frequently, and the vastness of the configuration space requires us to limit our search to sections of the configuration space. The flavor that we will be studying, is the search for the optimal configuration of atomic clusters, which is usually the configuration with the lowest energy.
In the first part of the project, we will introduce the concept of a "surrogate model". The purpose of this model, is to limit the amount of times we need to evaluate the black-box function. We will then introduce a variety of search methods, that we use to probe the surrogate model. These will be benchmarked against one another, on a simple 1d-function. In the second section, we apply these on a real physical system, using the AGOX-framework develop by Hammer and . In the final section, we evaluate the results of our search in the context of Hückle theory, which is a simple method that can be used to predict the energy of an atomic cluster.

Complexity of atomic clusters

Given some atomic cluster of N atoms, we wish to find the configuration of the atoms, that minimizes the energy of the system. We can immediately see, that the dimensionality of the system is equal to $3N$, corresponding to the possible directions each atom can move in. The dimensionality of the configuration space, is thereby a quickly growing function of the number of atoms. In addition to this, calculating the energy of an atomic cluster, is a complicated matter aswell. In order to get a precise answer, that enables us to distinguish between similar configurations, advanced methods like *density functional theory*. These methods have an enormous time complexity (source?) and can take multiple minutes, to calculate energy of even simple systems on a regular computer. In order to overcome this, we will now introduce *surrogate models*

Gaussian Process Regression

As described in the previous section, we make use of a surrogate model to give an intermediate view of the energy landscape. We construct this surrogate, using a method known as *gaussian process regression* (GPR). This method has the benefits of being parameter free, simple and the ability to interpolate data perfectly. In this section I will provide a brief overview of the method aswell as some examples. We start with some definitions,



2.0.1 *Gaussian Process*

A gaussian process is a special case of so-called random processes. A random process is an infinite collection of random variables $\{X_t, t \in T\}$, where T is some (continuous) index set. The way in which the X_T relate to one another, determines the type of random process. For gaussian processes the joint distribution of any finite subset $X_{1,\ldots,n}$ is a multivariate normal gaussian,

$$X_{1,\ldots,n} = (X_1, \ldots, X_n) \sim N(0, \Sigma).$$

Where Σ is the covariance matrix, which can be calculated using the GP's associated *kernel* K in the following way,

$$\Sigma_{ij} = K(X_i, X_j).$$

Where the kernel is some positive function. The behaviour of a GP, is therefore completely determined by the choice of kernel. A GP, can be viewed as a distribution over functions, where the functions domain is the index set T and its image, the possible realizations of the variables X_t. The general shape of the function, whether it is continuous, periodic and so forth, is determined by the kernel. A common choice of kernel, is the *radial basis function*

$$K_{RBF}(X_i, X_j) = \exp\left(-\frac{|X_i - X_j|^2}{\theta}\right),$$

which has the nice property of being infinitely differentiable. The constant θ is a hyperparameter, which is used to tune the kernel. A variety of sampled functions is shown in the FIG.

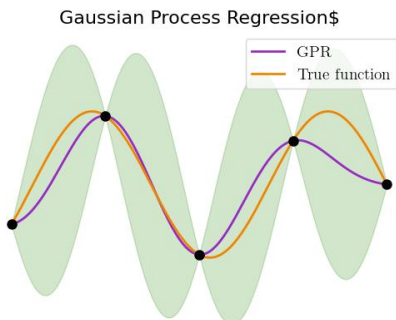
2.0.2 *Regression*

In general, we are not interested in sampling random functions. Instead we want to use our GP to make predictions. Let us assume that we want to model a real-valued function f, whose domain is an interval $I \subset \mathbb{R}$. Let $X = \{X_1, \ldots, X_n\}$ be our data points and $f(X)$ their "true" values. Our goal is to use this data to predict the value of f at a new point $\widetilde{x}$. The first step, is to construct the joint probability distribution,

$$(X, \widetilde{x}) \sim N\left(0, \begin{bmatrix} \Sigma_{XX} & \Sigma_{X\widetilde{x}} \\ \Sigma_{\widetilde{x}X} & \Sigma_{\widetilde{x}\widetilde{x}} \end{bmatrix}\right).$$

We can now condition on the true values of X, to obtain a conditional probability distribution for $\widetilde{x}$. I have chosen to leave out the derivation of this distribution as it is a bit cumbersome.

$$(X, \widetilde{x}) \mid X = f(X) \sim N\left(\mu(\widetilde{x}), \sigma^2(\widetilde{x})\right).$$



where the mean and variance are given by,

$$\mu\left(\widetilde{x}\right) = \left(\Sigma_{\widetilde{x}X}\Sigma_{XX}^{-1}\right) \cdot f\left(X\right)$$
$$\sigma^2\left(\widetilde{x}\right) = \Sigma_{\widetilde{x}\widetilde{x}} - \Sigma_{\widetilde{x}X}\Sigma_{XX}^{-1}\Sigma_{X\widetilde{x}}.$$

We have thus obtained a prediction and uncertainty for $f\left(\widetilde{x}\right)$ given our data. FIG illustrates aan example. An important part of GPR, is the choice of kernel and hyperparameters. Since these fully determine the functions that a GP is able to reproduce, it is essential that they are chosen wisely. There exist a variety of methods that accomplish this, which I shall not delve into.

Search Methods

3.0.1 *Exploration vs. Exploitation*

We have now constructed our surrogated model $\mathbf{G}$, which in theory should allow us to approximate the energy landscape. We still need to figure how to use G in the most effective way. The question is, given the knowledge we can obtain from G, which point $x \in \Omega$, should we evaluate in $\mathbf{B}$ and update our model. There are a variety of methods we can employ, some of which will be explored shortly, but they all have to balance the tug and pull of exploration and exploitation. An overly exploitative method, tends to search areas where the model is already confident. This leads to quite precise predictions, but runs the risk of overlooking important areas, where the model is currently. An explorative method, prefers not to dwell in areas where the model has low uncertainty. While this reduces the chance of overlooking important areas, the model is not encouraged to explore areas in depth, even though they might contain the global minimum. A good search method balancesthe two, knowing when to explore new areas, and when to dive in and exploit the surrogate model in a specific area.

3.0.2 *Pure Exploitation*

The simplest search method is to simply evaluate the point with the lowest predicted energy.

$$\mathbf{x}^+ = \arg\min_{\mathbf{x}\in\Omega} \mu_{\mathbf{G}}(\mathbf{x}).$$

This is a purely exploitative method, as it "assumes" that the model has perfect predictive power. This

3.0.3 *Lower Confidence Bound (LCB)*

The lower confidence bound method makes use of both mean and uncertainty provided by the GPR-model. It does this in a very simple way, by defining an aquisition function in the following way,

$$aqui\left(\mathbf{x}\right) = \mu_{\mathbf{G}}\left(\mathbf{x}\right) - \kappa\sigma_{\mathbf{G}}\left(\mathbf{x}\right).$$

Where κ is some constant that determines the emphasis put on the uncertainty, thereby controlling the relationship between exploration and exploitation. A large value of κ, will cause the second term to dominate, leading the search to favor unexplored areas ($\sigma_{\mathbf{G}}$ large). Conversely, as κ tends to zero, the LCB method approaches the purely exploitative method. This method has both clear advantages and downsides. It is unlikely to get stuck in a given area, as continued exploitation will cause the uncertainty to fall and theerby the aquisition function to rise. However, since it only cares about the sum of the mean and uncertainty, and not their ratios it is unable to determine the "likelihood" of a point being an improvement. This point is illustrated in fig.

$$\mathbf{x}^+ = \arg\max_{\mathbf{x}\in\Omega} \left\{\mu_{\mathbf{G}}(\mathbf{x}) - \kappa\sigma_{\mathbf{G}}(\mathbf{x})\right\}.$$

Expected Improvement

For our final search method, we proceed in the following way. First we note the currently best sampled point $y_{min} = \min(Y = \{y_1, \ldots, y_n\})$. This point will serve as the threshold we wish to overcome. Let $x \in \Omega$ be some point, we do not know $f(x)$, but given the GPR-model $\mathbf{G}$, we have access to an estimate of the value $\mu(x)$ and uncertainty $\sigma(x)$. We leverage these by modelling $f(x)$ with a normally distributed random variable A, with mean and spread given by $\mathbf{G}$. That is,

$$A \sim N(\mu, \sigma) \quad , \quad f_A(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

To measure the improvement at x, we introduce I, which we shall define as,

$$I[A(x)] = \max\{y_{min} - A(x), 0\}.$$

Given some realization of A, the improvement from y_{min} is at minimum zero, corresponding to A being greater than y_min. Otherwise, the improvement will be the distance A lies below y_{min}. We now wish to calculate the expected value of the improvement, which we call the expected improvement. Which can be done in the usual way using LOTUS,

$$\begin{aligned}\mathbb{E}[I] &= \int_{-\infty}^{\infty} I(z) f_A(z)\, dz \\ &= \int_{-\infty}^{\infty} \max\{y_{min} - z, 0\} f_A(z)\, dz \\ &= \int_{-\infty}^{y_{min}} (y_{min} - z) f_A(z)\, dz + 0.\end{aligned}$$

Substituting here, we are able to recover the standard normal distribtution,

$$= \int_{-\infty}^{\frac{y_{min}-\mu}{\sigma}} (y_{min} - r\mu - \sigma)\theta(r)\, dr.$$

By some fairly tedious integration by parts this can be shown to give,

$$\mathbb{E}[I] = (y_{min} - \mu)\Theta\left(\frac{y_{min}-\mu}{\sigma}\right) + \sigma\theta\left(\frac{y_{min}-\mu}{\sigma}\right).$$

where θ, Θ are the standard normal density and distribution functions. For each $x \in \Omega$ we can then calculate the expected improvement and select as our next point,

$$x^+ = \arg\max_{x\in\Omega} \mathbb{E}[I(x)].$$

Stated less formally, what we are doing is the following. For each point z below the threshold y_min, we calculate the improvement $I(z)$ and weight these with the probability $f_A(z)$. Keep in mind that $f_A(z)$ tends to zero, as z increases. The expected improvement will therefore be greater when f_A has more of its mass concentrated below y_{min}.

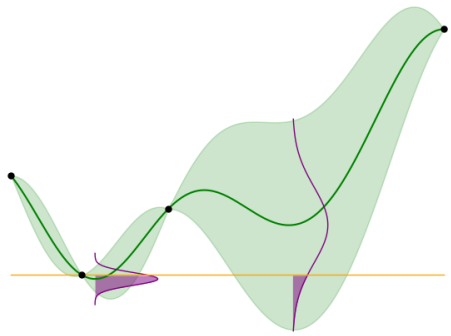


figure: Expected improvement calculated at two points. Notice, the second gaussian extends futher below the EI threshold than the first gaussian. However, the first gaussian has more of its mass concentrated below the threshold. The expected improvement is therefore greater at this point.

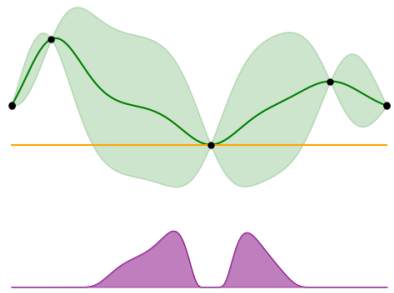


figure: EI shown for the entire space. Notice, that EI quickly approaches zero, as the surrogate model's uncertainty moves above the threshold.

4.0.1 *The δ parameter*

Our current algorithm, has a slight problem. In its current form, we are unable to steer the relationship between exploration and exploitation. We can remedy this by adding an additioal parameter δ. This parameter should be interpreted as moving the threshold we wish to beat,

$$I_\delta[A(x)] = \max\{(y_{min} - \delta) - A(x), 0\}.$$

Performing similar calculations as before, we can find the expected improvement to be,

$$\mathbb{E}[I] = ((y_{min} - \delta) - \mu)\,\Theta\left(\frac{(y_{min} - \delta) - \mu}{\sigma}\right) + \sigma\theta\left(\frac{(y_{min} - \delta) - \mu}{\sigma}\right).$$

The way the parameter δ affects the relationship between exploration and exploitation, can be illustrated with the following example. Let us assume we have a situation where,

$$G^{-1}(y_{min}) = \arg\min_{x\in\Omega}\{\mu(x)\}.$$

Corresponding to the situation, where the minimum sampled point is also the minimum of the surrogate model (As in figure [] in the previous section). In this case, points in the vicinity of $G^{-1}(y_{min})$, are almost guranteed to have positive expected improvement, as points x' in the vicinity will have values of $\mu(x')$ close to y_min. This implies that $f_A(x')$ has a lot its mass concentrated below y_min Therefore, the model will heavily favor exploitation. The addition of the parameter δ (moving the threshold down) penalizes points these points as they, in general, have low uncertainty. In practice, the δ is finetuned until a good balance between exploration and exploitation is found.

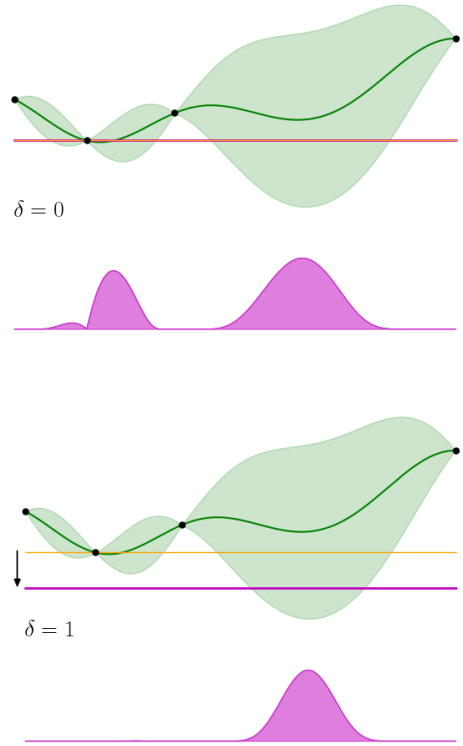


figure: Moving the threshold encourages exploration, as points in the vicinity of y_{min} are given low uncertainties by the GP. As a result, their probability distributions tend to be tightly concentrated around y_{min}.

Huckle Theory

In this final section of the report, we will attempt to evaluate the success of our search. What we aim to do is look at the structures we have found, and using the language of Huckle Theory, test whether their ranking in this paradigm is consistent with databased potential we have used.

5.0.1 *Method and Assumptions*

In the Huckle framework we assume that the cluster orbitals can be described as a linear combination of atomic orbitals (LCAO), we denote these atomic orbitals $\{\phi_i\}$. Let us assume that we are dealing with N atoms in our cluster and denote the i'th atom by A_i. The combined molecular/cluster orbitals $\{\Psi_i\}$ then become,

$$|\Psi_i\rangle = \sum_k c_k |\phi_k\rangle .$$

We can substitute this equation into the Schrödinger equation obtaining,

$$\hat{H}|\Psi_i\rangle = E|\Psi_i\rangle$$
$$\sum_k \hat{H}|\phi_k\rangle = \sum_k E c_k |\phi_k\rangle .$$

Now, bearing in mind that our goal is to find suitable values of E, we construct a series of N equations by multiplying both sides by $\langle\phi_i|$. The i'th equation then becomes,

$$\sum_k c_k \left(\left\langle \phi_i \middle| \hat{H} \middle| \phi_k \right\rangle - E \langle \phi_i \mid \phi_k \rangle \right) = 0.$$

In order to simplify, we use the familiar notation $H_{ik} := \langle\phi_i|\hat{H}|\phi_k\rangle$ (the matrix elements of the hamiltonian) and $S_{ik} := \langle\phi_i \mid \phi_k\rangle$ (the overlap integrals). We can gather these equations in the following matrix equation,

$$\begin{pmatrix} H_{11} - ES_{11} & \dots & H_{1n} - ES_{1n} \\ \vdots & \ddots & \vdots \\ H_{n1} - ES_{n1} & \vdots & H_{nn} - ES_{nn} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} = 0.$$

Now, in theory, the H_{ij} and S_{ij} are all known values that can be calculated and E is the only unknown. As we know from linear algebra matrix equations of the form $Ax = 0$, have non-trivial solutions so long as $\det(A) = 0$. We can therefore have to solve the equation,

$$\det\left(\overline{H} - E\overline{S}\right) = 0.$$

for E in order to obtain the orbital energies of the cluster. In practice the overlap integrals and matrix-elements of the hamiltonian can be quite cumbersome to calculate. In order to overcome this, simple Hückle theory employs the following approximations

Approximations/assumptions

1. The hamiltonian integrals are assumed to be,

$$H_{ij} = \begin{cases} \alpha & i = j \\ \beta & \text{when } A_i \text{ are } A_j \text{ are bonded} \\ 0 & \text{otherwise} \end{cases} .$$

The assumption here, is that the energy (H_{ii}) of an electron in an isolated orbital is α and that the energy (H_{ij} of interaction between electrons on bonded atoms is β. This can be viewed as an assumption of equal bond lenghts, wherever bonds occur. α and β are both assumed to be negative, with $\beta > \alpha$. Finally we assume, that electrons on non-bonded atoms do not interact)

2. The (spatial) overlap integrals are assumed to be,

$$S_{ij} = 0.$$

3. We assume that the total binding energy of the system is,

$$E_{tot} = \sum_{k}^{states} = n_k \epsilon_k.$$

In other words, the energy is the sum over the number of electrons in a given orbital multiplied by the energy of said orbital.

Note, that these assumptions disable us from making exact energy calculations. However, we should still be able to compare the predicted "energies" of distinct configurations. We are now ready to tackle a simple example.

5.0.2 *Simple example - Au3*

Lets try the method in practice on a simple system. Consider the $Au3$ system, and the two following simple configurations (figure). The first configuration is linear, and features bonds between the first and second atoms, and the second and third atoms. The second configuration is triangular, and has bonds between all the atoms. Now given the assumptions outlined in the previous sections, we can construct the following Huckle matrices,

$$\hat{H}_{tri} = \begin{pmatrix} \alpha & \beta & \beta \\ \beta & \alpha & \beta \\ \beta & \beta & \alpha \end{pmatrix} \quad \text{and} \quad \hat{H}_{lin} = \begin{pmatrix} \alpha & \beta & 0 \\ \beta & \alpha & \beta \\ 0 & \beta & \alpha \end{pmatrix}.$$

Using elementary linear algebra the eigenvalues of these matrices are found to be,

$$\epsilon_{tri} = \begin{cases} \alpha + \sqrt{2}\beta \\ \alpha \\ \alpha - \sqrt{2}\beta \end{cases} \qquad \epsilon_{lin} = \begin{cases} \alpha + 2\beta \\ \alpha - \beta \end{cases} \quad .$$

Now filling up the orbitals in the usual way, with 2 electrons at the lowest level and 1 at the second lowest, we obtain the following binding energies,

$$E_{lin} = 3\alpha + 2\sqrt{2}\beta \quad , \quad E_{lin} = 3\alpha + 3\beta.$$

Confirming my suspicion that the triangular has the lowest energy.

Comparison of DFT, EMT and Huckle

In this section we shall compare the results from our search using EMT and DFT with the predictions our Huckle model make.

What graph properties lead to low energy in Huckle?

In the Huckle framework, the predicted energy depends on the eigenvalues (spectrum) of the adjacency matrix. In the context of Au clusters, where each atom has a single valence electron, the predicted energy will be twice the sum of the lowest eigenvalues. In this section I will study some different properties of graphs, and try to determine how these relate to the dispersion of the eigenvalues.

7.0.1 *What we are interested in.*

Consider a graph G with n components and associated adjacency n-dimensional matrix A. Then, since A is symmetric and positive semi-definite it has real eigenvalues $\lambda_i \in \mathbb{R}$, which we shall order in the following way,

$$\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n.$$

Notice also that the symmetry of A implies that,

$$\sum_i \lambda_i = \mathrm{Tr}\,(A) = 0.$$

As a result of this there must be some form of balance in the eigenvalues, with the positive ones being balanced out by the negative ones. Since the systems we are considering have one valence electron pr. atom, the total energy of the system will be,

$$E = \sum_i^{n/2} 2 \cdot \lambda_i \quad n \text{ even}.$$

Keeping this in mind, graphs with extreme eigenvalues seem to be good candidates. The first kind of graph, that we shall consider are so-called bipartite graphs.

7.0.2 *Bipartite graphs*

A bipartite graph, is a graph that can be labeled with two colors, such that no connected vertices share the same color. An example of both a bipartite and non-bipartite graph is shown in fig. Qualitatively, bipartite graphs do not appear to be good candidates. This is due to the fact that they admit heavily fluctuating wavefunctions, which naturally have large eigenvalues.
Bipartite graphs also satisfy the following property,

$$\lambda_1 = -\lambda_n.$$

Proof. □