

PHYSICAL SCIENCES

Molecular docking with Gaussian Boson Sampling

Leonardo Banchi^{1*}, Mark Fingerhuth², Tomas Babej², Christopher Ing², Juan Miguel Arrazola¹

Gaussian Boson Samplers are photonic quantum devices with the potential to perform intractable tasks for classical systems. As with other near-term quantum technologies, an outstanding challenge is to identify specific problems of practical interest where these devices can prove useful. Here, we show that Gaussian Boson Samplers can be used to predict molecular docking configurations, a central problem for pharmaceutical drug design. We develop an approach where the problem is reduced to finding the maximum weighted clique in a graph, and show that Gaussian Boson Samplers can be programmed to sample large-weight cliques, i.e., stable docking configurations, with high probability, even with photon losses. We also describe how outputs from the device can be used to enhance the performance of classical algorithms. To benchmark our approach, we predict the binding mode of a ligand to the tumor necrosis factor- α converting enzyme, a target linked to immune system diseases and cancer.

INTRODUCTION

In his lecture “Simulating Physics with Computers” (1), Richard Feynman famously argued that classical computing techniques alone are insufficient to simulate quantum physics. Since then, substantial progress has been made in formalizing this intuition by finding explicit examples of quantum systems whose classical simulation can be convincingly shown to require exponential resources. An example is Boson Sampling, first introduced by Aaronson and Arkhipov (2). In this paradigm, identical photons interfere by passing through a network of beam splitters and phase shifters and are subsequently detected at the output ports of the network. Despite the simplicity of this model, it has been shown that, under standard complexity theoretic conjectures, generating samples from the output photon distribution requires exponential time on a classical computer (2–4). Several variants of Boson Sampling have been proposed that aim at decreasing the technical challenges with its experimental implementation (5–8).

Most efforts in the study of Boson Sampling have been focused on its viability to disprove the Extended Church-Turing thesis (9), not on its potential practical applications. Nevertheless, it is possible to ask: If Boson Sampling devices are powerful enough that they cannot be simulated with conventional computers, is there a way of programming them to perform a useful task? Practical applications of Boson Sampling have already been reported. In (10), it was shown that a Boson Sampling device can be used to efficiently estimate the vibronic spectra of molecules, a problem for which, in general, no efficient algorithm is known. Proof-of-principle demonstrations have also been reported (11, 12). In addition, (13–15) discuss how a specific model known as Gaussian Boson Sampling (GBS) can be used in combinatorial optimization problems concerned with identifying large clusters of data.

Molecular docking is a computational method for predicting the optimal interaction of two molecules, typically a small-molecule ligand and a target receptor. This method works by searching the configurational space of the two molecules and scoring each pose using a potential energy function. Using molecular structures to determine stable ligand-receptor complexes is a central problem in pharma-

ceutical drug design (16, 17). Several techniques for finding stable ligand-receptor configurations have been developed, including shape-complementarity methods (18–21) and molecular simulation of the ligand-receptor interactions (22), which vary in their computational requirements. For high-throughput virtual screening of large chemical libraries, it is desirable to search and score ligand-receptor configurations using as few computational resources as possible (23). However, it is important to keep in mind that accurate docking in the absence of a reliable scoring function is a major challenge. Molecular docking is related to the prediction of molecular similarity for ligand-based virtual screening, which has been formulated as a graph theoretic problem for quantum annealers (24, 25).

In this work, we show that GBS can be used to find docking configurations between ligands and receptors. We extend the binding interaction graph approach, where the problem of identifying docking configurations can be reduced to finding large clusters in weighted graphs (26, 27). We then show how GBS devices can be programmed to sample from distributions that assign large probabilities to these clusters, thus helping in their identification. The best docking configurations are selected based on the weight of the corresponding clique. GBS devices therefore enhance the search for stable configurations—which correspond to cliques—while the scoring function is implicit in the weights of the input graph. Docking configurations can be obtained by direct sampling or by hybrid algorithms where the GBS outputs are post-processed using classical techniques. The goal of this paper is to demonstrate that GBS devices can improve the sampling of docking configurations, whereas it does not address the difficulties associated with scoring functions (28, 29), which represent the major hurdle in accurate docking. We apply our method to find molecular docking configurations for a known ligand-receptor interaction (30), using a simplified representation to make the problem solvable through numerical simulations. Several therapeutic agents targeting this protein have entered into clinical trials for both cancer and inflammatory diseases (31). A different protein structure is studied in the Supplementary Materials.

RESULTS

Before presenting our results, we provide relevant background information on molecular docking. Readers not familiar with GBS may read Materials and Methods.

Copyright © 2020
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim to
original U.S. Government
Works. Distributed
under a Creative
Commons Attribution
License 4.0 (CC BY).

¹Xanadu, 372 Richmond St W, Toronto, ON M5V 1X6, Canada. ²ProteinQure Inc., 192 Spadina Ave, Toronto, ON M5T 2C2, Canada.

*Corresponding author. Email: banchi.leonardo@gmail.com

Molecular docking

Molecular docking is a computational tool for rational structure-based drug discovery. Docking algorithms predict noncovalent interactions between a drug molecule (ligand) and a target macromolecule (receptor) starting from unbound three-dimensional structures of both components. The output of such algorithms is predicted three-dimensional orientations of the ligand with respect to the receptor binding site and the respective score for each orientation. Reliable determination of the most probable ligand orientation, and its ranking within a series of compounds, requires accurate scoring functions and efficient search algorithms (28). The scoring function contains a collection of physical or empirical parameters that are sufficient to score binding orientation and interactions in agreement with experimentally determined data on active and inactive ligands. The search algorithm describes an optimization approach that can be used to obtain the minimum of a scoring function, typically by scanning across translational and rotational degrees of freedom of the ligand in the chemical environment of the receptor. In the simplest case, both the ligand and the receptor can be approximated as rigid bodies, but more accurate methods can account for inherent flexibility of the ligand and receptor (21). As is the case for most molecular modeling approaches, a trade-off exists between accuracy and speed.

High-performance algorithms enable molecular docking to be used for screening large compound libraries against one or more protein targets. Molecular docking and structure-based virtual screening are routinely used in pharmaceutical research and development (32). However, evaluating billions of compounds requires accurate and computationally efficient algorithms for binding pose prediction. Widely used approaches for molecular docking use heuristic search methods [simulated annealing (33) and evolutionary algorithms (34)] and deterministic methods (35). In one combinatorial formulation of the binding problem used in the DOCK 4.0, FLOG, and SQ algorithms, an isomorphous subgraph matching method is used to generate ligand orientations in the binding site (26, 36–38). In this formulation of the binding problem, both the ligand and the binding site of the receptor are represented as complete graphs. The vertices of these graphs are points that define molecular geometry, and edges capture the Euclidean distance between these points. To strike a balance between the expressiveness of the graph and its size, we reduce the all-atom molecular models of the ligand and receptor to a pharmacophore representation (39).

A pharmacophore is a set of points that have a large influence on the molecule's pharmacological and biological interactions. These points may define a common subset of features, such as charged chemical groups or hydrophobic regions, that may be shared across a larger group of active compounds. For the purposes of this study, we define six different types of pharmacophore points: negative/positive charge, hydrogen-bond donor/acceptor, hydrophobe, and aromatic ring. In the graph representation, the type of the pharmacophore point is preserved as a label associated with its vertex. Hence, we refer to this molecular graph representation as a labeled distance graph (see also section S2). As illustrated in Fig. 1, a labeled distance graph is constructed as follows for both the ligand and receptor:

- 1) Heuristically identify pharmacophore points likely to be involved in the binding interaction. These form the vertices of the graph.
- 2) Add an edge between every pair of vertices and set its weight to the Euclidean distance between the pharmacophore points they represent.

- 3) Assign a label to every vertex according to the respective type of pharmacophore point it represents.

Mapping molecular docking to maximum weighted clique

The labeled distance graphs described in the previous section capture the geometric three-dimensional shapes and the molecular features of both the protein binding site and the ligand that interacts with it. In this section, akin to (26), we combine these two graphs into a single binding interaction graph. Subsequently, we reduce the molecular docking problem to the problem of finding the maximum weighted clique.

If two pharmacophore points are interacting, they form a contact. A binding pose can be defined by a set of three or more contacts that are not colinear. We model contacts as pairs of interacting vertices of the labeled distance graphs of the ligand and the binding site. Consider the labeled distance graph G_L of the ligand and the labeled distance graph G_B of the binding site, with their vertex sets V_L and V_B , respectively. A contact is then represented by a single vertex $c_i \in V_L \times V_B$. The set of possible contacts forms the vertices of the binding interaction graph. In principle, any pharmacophore point of the ligand could be interacting with any pharmacophore point of the binding site, and therefore, we have to consider every possible pair of corresponding interacting vertices. Hence, the number of vertices of the binding interaction graph is nm , where n is the number of vertices of the labeled distance graph G_L and m is the number of vertices of the labeled distance graph G_B .

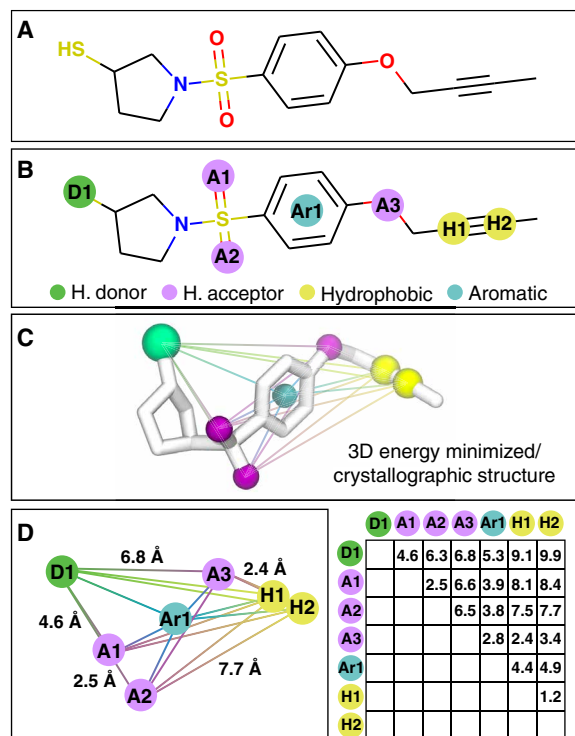


Fig. 1. Construction of the labeled distance graph for a ligand molecule. (A) Planar structure of the ligand molecule. Pharmacophore points of the molecule (B) are identified, and their pairwise distance is measured using the known three-dimensional (3D) structure (C). This information is combined in the labeled distance graph for the ligand molecule (D), where vertices represent the pharmacophore points and edge weights of their respective pairwise distance [the complete weight matrix is on the right of (D)].

The goal of the binding interaction graph is to model possible binding poses via sets of contacts. However, not every combination of contacts is physically realizable. Two contacts are not being compatible if their mutual realization would violate the geometrical shapes of the ligand and the binding site. To model this, the binding interaction graph contains an edge between two contacts if and only if they are compatible. As a result, a pairwise compatible set of contacts, i.e., such as would arise from a true binding pose, forms a complete subgraph of the binding interaction graph. A complete subgraph, also called a clique, in a graph G is a subgraph where all possible pairs of vertices are connected by an edge.

The compatibility of contacts is captured by the notion of τ flexibility, which is illustrated in Fig. 2 (see also section S2). Although both the ligand and the binding site can exhibit a certain amount of flexibility, in general, geometric distances between two contacts have to be approximately the same on both the ligand and the binding site. Two contacts (v_{l1}, v_{b1}) and (v_{l2}, v_{b2}) form a τ flexible contact pair if the distance between the pharmacophore points on the ligand (points corresponding to vertices v_{l1} and v_{l2}) and the distance between the pharmacophore points on the binding site (points corresponding to

vertices v_{b1} and v_{b2}) do not differ by more than $\tau + 2\epsilon$ (see Fig. 2C). The constants τ and ϵ describe the flexibility constant and interaction distance, respectively. The role of τ and ϵ is therefore to determine which edges actually appear in the binding interaction graph.

To model varying interaction strengths between different types of pharmacophore points, we associate a different weight to every vertex of the binding interaction graph. The weights are derived using the pharmacophore labels that are captured in the labeled distance graphs of the ligand and the binding site. Given a set of labels \mathbb{L} , a potential function $\kappa: \mathbb{L} \times \mathbb{L} \rightarrow \mathbb{R}$ is applied to compute the weights of the individual vertices. This allows us to bias the algorithm toward stronger intermolecular interactions. Potential functions can be derived in several ways, ranging from pure data-based approaches such as statistical or knowledge-based potentials (40–42) to quantum-mechanical potentials (16). Details of the potential used in this study are described in “Numerical results.”

Hence, under the model derived in this study, the most likely binding poses correspond to vertex-heaviest cliques in the binding interaction graph. The problem of finding a maximum weighted clique is a generalization of the maximum clique problem of finding the clique with the maximum number of vertices. When G has n vertices, the number of possible subgraphs is $\mathcal{O}(2^n)$, so a brute force approach becomes rapidly infeasible for growing values of n . The max clique decision problem is NP-hard (43): As such, unless $P = NP$, in the worst case, any exact algorithm runs for superpolynomial time before finding the solution. There are deterministic and stochastic classical algorithms for finding both the maximum cliques and maximum weighted cliques or for finding good approximations when n is large (44).

It is crucial to understand that the outputs of this method are only as good as the quality of the binding interaction graph as a model for the docking problem. Graphs built exclusively from subsets of pharmacophores that are not involved in the correct binding pose will lead to incorrect pose predictions. Conversely, by construction, the largest weighted clique in accurate graphs will correspond to the correct binding pose. It is a feature of our approach that different values of the parameters τ and ϵ result in the same binding interaction graph and hence the correct maximum weighted clique, as shown in table S2. Following our procedure, the predicted binding configuration can be scored and ranked among a set of alternate molecules, or a continuous-space representation of the complex can be derived using distance restraints derived from the solution using an industry-standard docking algorithm. Our capability to couple conformational search and scoring through parameterization by weights of the graph suggests that we may not be subject to bottlenecks arising from scoring function evaluation. We see our approach as synergistic with parallelized continuous space methods, such as restrained docking or alternate hybrid protocols.

Max weighted clique from GBS

In this section, we show that a GBS device can be programmed to sample from a distribution that outputs the max weighted clique with high probability. The main technical challenge is to program a GBS device to sample, with high probability, subgraphs with a large total weight that are as close as possible to a clique. As shown in Materials and Methods and depicted in Fig. 3, a GBS device is composed of two main parts. In the first part, a chosen quantum Gaussian state is generated via squeezing, phase shifters, and beam splitters. The chosen state is identified by an $M \times M$ matrix that is related to the covariance matrix of the Gaussian state, where M is the number

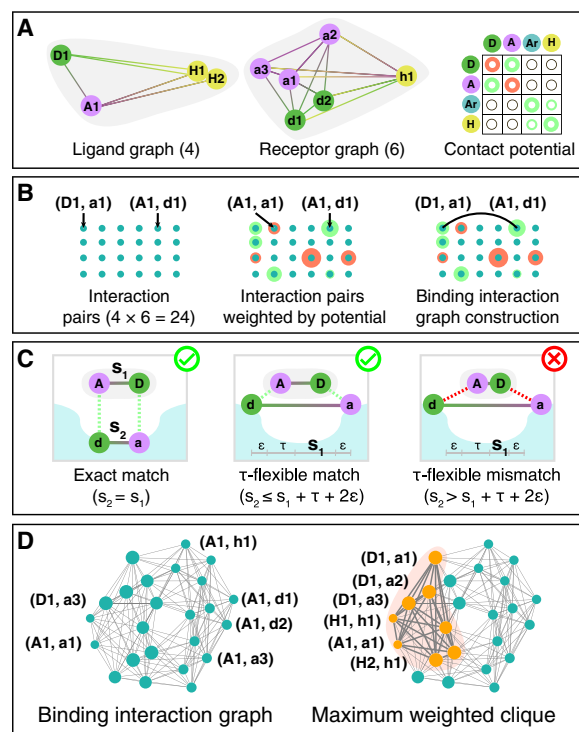


Fig. 2. Construction of the binding interaction graph. (A) Inputs for the construction of the binding interaction graph—two labeled graphs (one for the ligand and one for the receptor) and corresponding contact potential that captures the interaction strength between different types of vertex labels. We denote vertices on the ligand and receptor with uppercase and lowercase letters, respectively. The binding interaction graph is constructed (B) by creating a vertex for each possible contact between ligand and the receptor weighted by the contact potential. Pairs of vertices that represent compatible contacts [see (C) for various scenarios] are connected by an edge. The resulting graph is then used to search for potential binding poses (D). These are represented as complete subgraphs—also called cliques—of the graph, as they form a set of pairwise compatible contacts. The heaviest vertex-weighted cliques represent the most likely binding poses (maximum vertex-weighted clique depicted in orange).

of optical modes. The second part is made by photon-counting detectors that measure the number of photons coming out of each mode. To find a suitable input matrix for GBS, consider a graph with M vertices and with graph Laplacian $L = D - A$, where D is the degree matrix and A is the adjacency matrix. The normalized Laplacian (45) $\tilde{L} = D^{-1/2} L D^{-1/2}$ is positive semidefinite, and its spectrum is contained in $[0, 2]$. More generally, we define a rescaled matrix

$$B = \Omega(D - A)\Omega \quad (1)$$

where Ω is a suitable diagonal matrix. If the largest entry of Ω is bounded as shown in section S1, then the spectrum of B is contained in $[0, c]$, where $c \leq 1$ can be tuned depending on the maximum amount of squeezing obtainable experimentally. Using the decoupling theorem from section S1, we find that a GBS device can be programmed to sample from the distribution

$$p(S) \propto [\det(\Omega_S) \text{Haf}(A_S)]^2 \quad (2)$$

where Haf refers to the matrix Hafnian and we consider outputs $S = (n_1, \dots, n_M)$ with n_j detected photons in mode j , so $N = \sum_j n_j$ is the total number of photons. When we focus on the collision-free subspace, where $n_j \leq 1$, the dependence on the diagonal matrix D disappears so we may focus on programming GBS with a rescaled adjacency matrix $\Omega A \Omega$. As shown in Fig. 3, from a GBS sample S ,

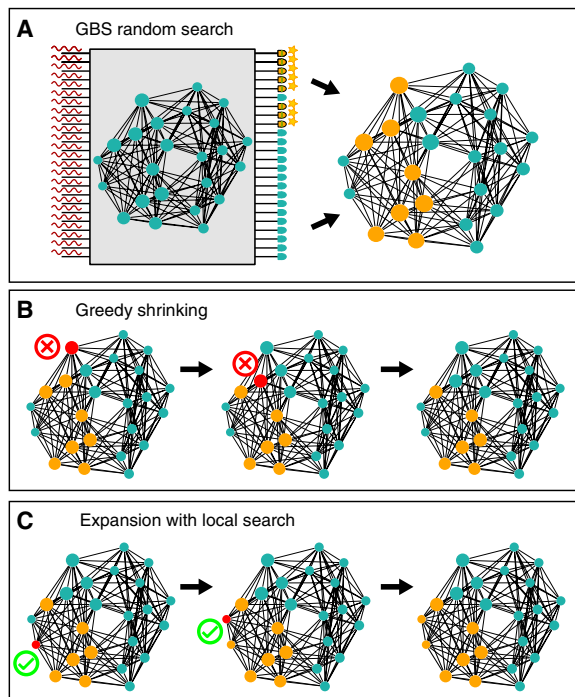


Fig. 3. Schematics of the protocol. Squeezed light is injected from the left into a GBS device, which is programmed to sample from a vertex-weighted graph. The presence (star) or absence of photons is measured by detectors on the right. GBS random search (A): On the basis of the ports where photons have been detected, we construct a subgraph (yellow vertices and dark edges) and check if it is a clique. If it is not a clique, greedy shrinking (B) iteratively removes a vertex (red node with a cross) until the remaining ones form a clique. Two shrinking iterations are shown in (B) from left to right. In local search (C), the found clique is expanded by iteratively adding, as long as possible, a neighboring vertex (red node with a tick) to get a bigger clique.

we construct the subgraph H of G made by vertices j with $n_j = 1$. The matrix A_S is the $N \times N$ adjacency matrix of H . The Hafnian of an adjacency matrix is maximum for the complete graph, namely, when H is a clique. Therefore, for a fixed total number of photons N , the Hafnian term maximizes the probability of detecting photon configurations that correspond to a clique.

Different choices are possible for the weighting matrix Ω . For an unweighted graph, convenient choices are either a constant Ω or $\Omega \propto D$. In the former case, $\det \Omega_S = c^N$ for $c < 1$, so the parameter c can be tuned by squeezing to penalize larger N , i.e., larger subgraphs (see section S1.3). In the latter case, $\det \Omega = c^N \det D$ is proportional to the Narumi-Katayama index (46), which describes some topological properties of the graph. Similarly to the Hafnian, it is maximum when H is a clique.

For a vertex-weighted graph, we can use the freedom of choosing Ω to favor subgraphs with larger total weight. There are multiple ways of introducing the weights w_j in Ω and a convenient choice is

$$\Omega_{ii} = c(1 + \alpha w_i) \quad (3)$$

where c is a normalization to ensure the correct spectral properties and $\alpha > 0$ is a constant. When α is small, the determinant term $\det \Omega_S \approx 1 + \alpha \sum_{j:n_j=1} w_j$ is large when the subgraph H has a large total weight. This is useful for the max weighted clique problem as it introduces a useful bias in the GBS probability of Eq. 2 that favors heavier subgraphs. However, if α is too large, the Hafnian term in Eq. 2 becomes less important and GBS will sample heavy subgraphs that typically do not contain cliques. To prevent this occurrence, the parameter α must be chosen carefully. Ideally, the weights should give a positive bias to heavy cliques but should not favor heavy subgraphs that are not cliques. More details are discussed in section S1.

Hybrid algorithms

GBS devices can, in principle, have a very high sampling rate—primarily limited by detector dead time—so just by observing the photon distribution, it is possible to extract the maximum weighted clique for small enough graphs. We call this simple strategy GBS random search—see Fig. 3 for a graphical explanation of the method. However, selecting photon outcomes that correspond only to cliques means wasting samples that are potentially close to the solution. An optimally programmed GBS device will sample from both the correct solution and neighboring configurations with high probability. Therefore, we propose two algorithms to post-process all GBS data that incur an overhead in run time but are especially useful for finding cliques in larger graphs.

Greedy shrinking

Starting from an output subgraph H from GBS, vertices are removed based on a local rule until a clique is found—see Fig. 3 for a graphical explanation of the method. Removal is based on vertex degree and weight. Vertices with small degree are unlikely to be part of a clique, making them good candidates to be discarded. The role of the weights is less straightforward: Vertices with low weight may not be part of the max weighted clique, but this assumption may be incorrect if the clique is made by a heavy core together with a few light vertices. Because of this, vertex degree is prioritized over vertex weight during the greedy shrinking stage. More precisely, the algorithm proceeds as follows:

1) From a GBS outcome, build a subgraph H with vertices corresponding to the detectors that “click.”

- 2) If H is a clique, return H .
- 3) Otherwise, set v as the set of vertices in H with smallest degree.
- 4) Set w as the subset of v with lowest weight.
- 5) Remove a random element of w from H and go back to step 2.

Expansion with local search

GBS provides high-rate samples from max cliques, and greedy shrinking enhances the probability of finding a solution via classical post-processing of sampled configurations. We may increase the probability of finding the solution even further at the cost of a few more classical steps. This is done by using a local search algorithm that tries to expand the clique with neighboring vertices, as shown also in Fig. 3. We use algorithms such as dynamic local search (DLS) (47) and phased local search (PLS) (48) that are among the best-performing classical algorithms for max clique (44). These algorithms usually start with a candidate clique formed by a single random vertex and then try to expand the clique size and replace some of its vertices by locally exploring the neighborhood. More precisely, the following iteration is repeated until a sufficiently good solution is found, or the maximal number of steps is reached:

1) Grow stage: Starting from a given clique, generate the set of vertices that are connected to all vertices in the clique. If this set is nonempty, select one vertex at random, possibly with large weight, and add it to the clique.

2) Swap stage: If the above set is empty, generate the set of vertices that are connected to all vertices in the clique except one (say v). From this new set, select a vertex at random and swap it with v . This gives a new clique of the same size but with different vertices, thus constituting a local change to the clique. For max weighted clique, the swapping rule also considers vertex weight.

An important aspect of the above local search is that, at each iteration step, the candidate solution is always a clique and the algorithm tries to expand it as much as possible. GBS can be included in this strategy in view of its ability to provide a starting configuration that is not a mere random vertex. A GBS output after greedy shrinking is always a clique, with a comparatively large probability of being close to the maximum clique. In case the candidate output from greedy shrinking is not the maximum clique, then it can be expanded with a few iterations of local search. Because the cliques sampled from a carefully programmed GBS device are, with high probability, larger than just a random vertex, the number of classical expansion steps is expected to be significantly reduced. This will be demonstrated with relevant numerical examples in the following section.

Numerical results

We study the binding interaction between the tumor necrosis factor- α converting enzyme (TACE) and a thiol-containing aryl sulfonamide compound (AS). A different protein structure is studied in section S4.1. TACE was chosen because of the planar geometry of the active site cleft and its high relevance to the pharmaceutical industry. Because of its role in the release of membrane-anchored cytokines like the tumor necrosis factor- α , it is a promising drug target for the treatment of certain types of cancer, Crohn's disease, and rheumatoid arthritis (31). The ligand under consideration is part of a series of thiol-containing aryl sulfonamides that exhibit potent inhibition of TACE, with a binding pose supported by a crystallographic structure (30). This complex provides an important testbed to benchmark our GBS-enhanced method. We use a coarse-grained representation of the TACE-AS complex that we describe below.

Coarse graining is used to reduce the dimensionality of the binding interaction graph to simulate GBS with a classical computer but may be avoided when using a physical quantum device with the suitable number of modes. As we will show, our method is able to find the correct binding pose without requiring all-atom representation or simulation of the ligand/receptor complex.

The binding interaction graph for the TACE-AS complex is constructed by first extracting all the pharmacophore points on ligand and receptor using the software package rdkit (49). To simplify numerical simulations, we identify the relevant pairs of pharmacophore points on the ligand and receptor that are within a distance of 4 Å of each other, and whose label pairs are either hydrogen donor/acceptor, hydrophobe/hydrophobe, negative/positive charge, or aromatic/aromatic. After this procedure, we get four nodes on the ligand and six nodes on the receptor and create two labeled distance graphs as illustrated in Fig. 1. Note that the only way for two different proteins to yield the same graph is for both of them to include the same pharmacophores, all located at the same distance from each other. The knowledge-based potential is derived by combining information from PDBbind (50), a curated dataset of protein-ligand interactions, and the Drugscore potential (51). More details are presented in section S3, the resulting knowledge-based potential is shown in table S1, and the stability of the generated graphs for different values of τ is discussed in table S2.

Using this knowledge-based potential, we combine the two labeled distance graphs into the TACE-AS binding interaction graph, as shown in Fig. 2. A summary of our graph-based molecular docking approach is shown in Fig. 4, which includes a molecular rendering of the predicted binding interactions of the AS ligand in the TACE binding site using the crystallographic structure of this complex (Protein Data Bank: 2OI0) (30). These interactions correspond to the maximum vertex-weighted clique in the TACE-AS graph. This set of pharmacophore interactions can be used as constraints in a subsequent round of molecular docking to deduce three-dimensional structures of the ligand-receptor complex (52, 53). We now study the search of the maximum weighted clique on the TACE-AS graph via a hierarchy of algorithms in increasing order of sophistication. As discussed previously, these are the following:

- 1) Random search: Generate subgraphs at random and pick the cliques with the largest weight among the outputs.
- 2) Greedy shrinking: Generate a large random subgraph and remove vertices until a clique is obtained. Vertices are removed by taking into account both their degree and their weight.
- 3) Shrinking + local search: Use the output of the greedy shrinking algorithm as the input to a DLS/PLS local search algorithm (44).

These form a hierarchy in the sense that random search is a subroutine of greedy shrinking, which is itself a subroutine of shrinking + local search. For each of these algorithms, we compare the performance of standard classical strategies with their quantum-classical hybrid versions introduced in the previous sections, where the random subgraph is sampled via GBS. We remark that our classical strategies are based on PLS/DLS, which are among the best-performing classical algorithms for max clique (44). Moreover, for a fair comparison with GBS-based approaches, the classical data are generated as follows: We first sample a subgraph size N from a normal distribution with the same mean $\langle N \rangle$ and variance ΔN^2 as the GBS distribution, then uniformly generate a random subgraph with size N .

We begin our analysis with a pure GBS random search. We consider GBS with threshold detectors, which register measurement outcomes

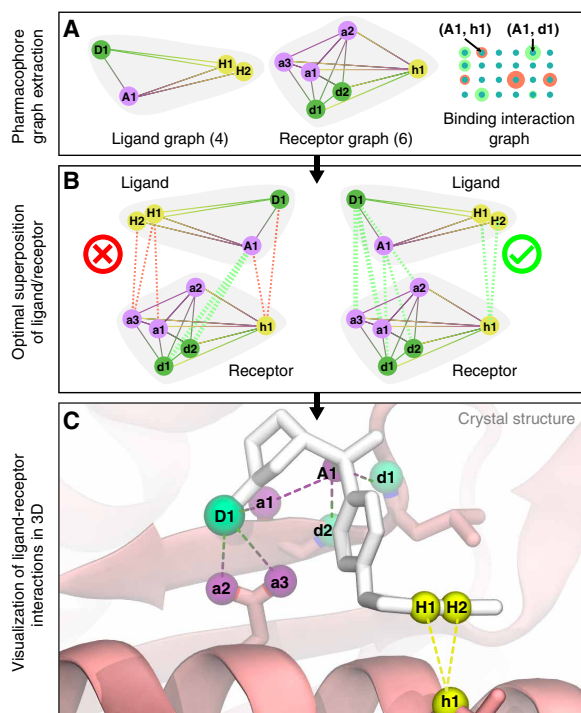


Fig. 4. Graph-based molecular docking of an aryl sulfonamide compound to TACE. (A) Two labeled distance graphs—one for the aryl sulfonamide compound and one for the TACE receptor—and the resulting TACE-AS binding interaction graph. Construction of the labeled distance graph and binding interaction graph are described in Figs. 1 and 2. Pharmacophore points on the ligand and receptor are labeled with uppercase and lowercase letters, respectively. The search for the maximum vertex-weighted clique within the TACE-AS graph is illustrated in (B). Each clique in the TACE-AS graph corresponds to a different superposition of the ligand molecule and the TACE receptor. The correct ligand-receptor superposition corresponding to the maximum weighted clique in the TACE-AS graph is shown on the right. (C) Crystallographic structure of the TACE-AS complex with optimal ligand-receptor interactions correctly predicted by the maximum weighted clique. We omit the metal cofactor in the enzyme active site for visual clarity, as it was not considered as a pharmacophore point under our procedure.

as either “no-click” (absence of photons) or “click” (presence of one or more photons). We use either a brute force approach to calculate the resulting probability distribution or, when that becomes infeasible, the exact sampling algorithm discussed in (54). Given the complexity of simulating GBS with classical computers, for simplicity in numerical benchmarking, we first consider the simpler case where the maximum clique size is known, so we can post-select GBS data to have a fixed number of detection clicks. This drastically simplifies numerical simulations (see section S1.4 for details) at the expense of disregarding data that would otherwise be present in an experimental setting. A similar reduction is applied to classical data for fair comparison.

For the TACE-AS binding interaction graph, the largest and heaviest cliques both have eight vertices, so we fix $N = 8$. There are a total of 19 cliques of this size in the graph (see also fig. S1 in section S4). In Fig. 5, we show the outcomes of a numerical experiment where a GBS device has been programmed to sample from the Hafnian of $\Omega\Lambda\Omega$, with Ω as in Eq. 3. For simplicity, we choose $\alpha = 1$ in Eq. 3, although performance can be slightly improved with optimized values of α . On the other hand, the parameter c does not play any role

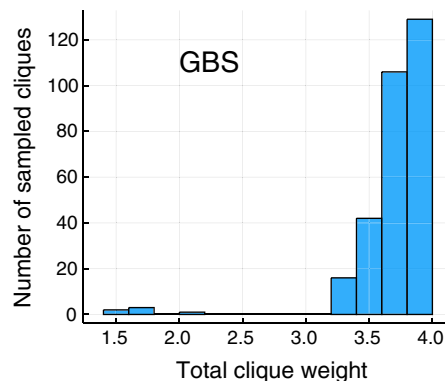


Fig. 5. GBS random search sampling rate. Number of cliques sampled from a GBS device as a function of the total clique weight $\sum_{j \in C} w_j$. The GBS output has been post-selected to 10^5 samples with total number of detector clicks $N = 8$. With the same number of samples (each with $N = 8$ nodes), classical random search only found three cliques (not shown), none of them with maximum weight.

in the post-selected data, but it does change the overall probability of getting samples of size $N = 8$. For comparison, we have also studied a purely classical random search, where each datum is a uniform random subgraph with N vertices. We observe only three cliques over 10^5 samples. On the other hand, as shown in Fig. 5, GBS is able to produce roughly 300 cliques directly from sampling, without any classical post-processing. This indicates that the GBS distribution is favoring cliques with large weights, as intended.

Post-selecting on the number of detector clicks is an unwise strategy when using real GBS devices because it disregards otherwise useful samples. Moreover, the size of the maximum weighted clique is generally unknown. Therefore, we now use the shrinking strategy discussed in the previous sections to extract a clique from every sample, without any post-selection.

In Fig. 6, we study the performance of greedy shrinking with GBS data. These data consist of 10^4 samples obtained from an exact numerical sampling algorithm (54). Each sample corresponds to a subgraph and, unlike Fig. 5, here, any subgraph size is considered. These results show that with GBS and greedy shrinking—a simple classical post-processing heuristic—it is possible to obtain the maximum weighted clique with sufficiently high probability. The histogram in Fig. 6 has a sharp peak corresponding to the clique of maximum size $N = 8$ and maximum weight ≈ 3.99 . The success rate in sampling from the max weighted clique is $\approx 12\%$, and the overall sampling rate for $N = 8$ cliques is $\approx 19\%$. Greedy shrinking with purely classical random data is shown in fig. S2. Although the classical distribution is chosen to have the same mean and variance as the GBS distribution, its performance is considerably worse: The maximum weighted clique is obtained only 1% of the time compared to 12% for GBS. This shows that GBS with greedy shrinking is already able to find the maximum weight clique of the graph after only a few repetitions.

Last, we study how the cliques obtained from GBS with greedy shrinking can be enlarged or improved via local search. Figure 7 shows the performance of the hybrid GBS shrinking + local search algorithm compared to a classical strategy. The results indicate that GBS not only provides better initial estimates after greedy shrinking (zero iteration steps) but also maintains a significant margin compared to classical strategies as the number of steps is increased. After $k = 8$ local expansion steps, the probability of finding the maximum

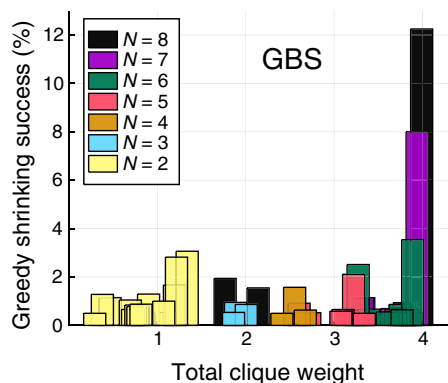


Fig. 6. Greedy shrinking success rate. Success rate in finding cliques of different sizes ($N = 2, \dots, N_{\max}$), when the max clique has size $N_{\max} = 8$, as a function of the total clique weight $\sum_{j \in C} w_j$. We used greedy shrinking over 10^4 GBS samples, ignoring trivial zero-photon outcomes. Outcomes with low ($<0.5\%$) success rate are not shown.

weighted clique is as high as 60%, while the classical strategy has a considerably smaller success rate of $<30\%$. After many steps, the success rate saturates: Using GBS, the success rate gets close to 70%, while for the purely classical approach it remains under approximately 35%. The latter performance is also achieved by a “vanilla” DLS/PLS classical algorithm (44), where each initial configuration is a single random vertex. This shows that sampling and shrinking do not help in a fully classical strategy, whereas an advantage is observed in the quantum hybrid approach with initial GBS samples, which are expected to be closer to the real solution.

The role of noise and squeezing is discussed in section S4, where we show that GBS success rate is not diminished by the effect of noise provided that the amount of squeezing is increased accordingly. Therefore, GBS shrinking and its variant with local search are robust against noise, maintaining a significant margin compared to purely classical strategies.

DISCUSSION

We have shown that GBS can be used to predict accurate molecular docking configurations, a central problem in pharmaceutical research and development. This is achieved by first mapping the docking problem to the task of finding large cliques in a vertex-weighted graph, then programming the GBS device to sample these cliques with high probability. This constitutes an example of the viability of near-term quantum photonic devices to tackle problems of practical interest. Further study is required to quantify the impact of improved max clique detection on the overall performance of molecular docking and to verify the applicability of this method across diverse clinically relevant ligand-receptor complexes.

Established algorithms for obtaining molecular docking configurations exist, but industrial drug discovery workflows routinely require large-scale virtual screening to enrich chemical libraries for lead candidates. In this case, a fast method for predicting docking configurations is essential. In principle, photonic devices such as Gaussian Boson Samplers can operate at very high rates and may potentially provide solutions in shorter timeframes. In addition, by sampling better random subgraphs, GBS serves as a technique to enhance the performance of classical algorithms because it increases

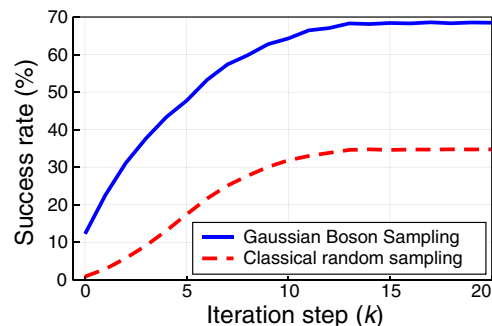


Fig. 7. GBS versus classical success rate. Success rate in finding the maximum weighted clique after greedy shrinking and k expansion steps with local search. Samples are generated from either GBS or a purely classical approach. GBS maintains a significantly higher success rate over all iteration steps.

the success rate of identifying large weighted cliques. This property is relevant and applicable in any context where identifying clusters in graphs is important beyond applications in molecular docking.

More broadly, our results establish a connection between seemingly disparate physical systems: The statistical properties of photons interacting in a linear-optical network can encode information about the spatial configuration of molecules when they combine to form larger complexes. In other words, we have found that when the interaction between fundamental particles is carefully engineered, they acquire collective properties that can be probed to perform useful tasks. A complete understanding of the capabilities of emerging quantum technologies may thus require further exploration of systems that, even if incapable of universal quantum computation, can still be programmed to exhibit properties that can be harnessed for practical applications.

MATERIALS AND METHODS

Gaussian Boson Sampling

Quantum systems such as the quantum harmonic oscillator or the quantized electromagnetic field can be described by phase-space methods. Here, each state is uniquely determined by a quasi-probability distribution such as the Wigner function $W(x, p)$ over its position x and momentum p variables. A quantum state is called Gaussian if its Wigner function is Gaussian (55). Any multimode Gaussian state ρ is parametrized by its first and second moments, namely, the displacement $\alpha_j = \text{Tr}[\rho \hat{\xi}_j]$ and the covariance matrix σ with entries $\sigma_{jk} = \text{Tr}[\{\hat{\xi}_j, \hat{\xi}_k\}]/2$, where $\hat{\xi}_j$ is a vector of creation and annihilation operators: calling M the number of modes, $\hat{\xi}_j = \hat{a}_j = (\hat{x}_j + i\hat{p}_j)/\sqrt{2}$ and $\hat{\xi}_{M+j} = \hat{a}_j^\dagger$ for $j = 1, \dots, M$. Gaussian quantum states are ubiquitous in quantum optics and have enabled detailed theoretical modeling and coherent manipulations in experiments (55).

In spite of their infinite-dimensional Hilbert space, Gaussian states can be simulated efficiently, as their evolution can be modeled by linear transformations such as Bogoliubov rotations (56). However, when non-Gaussian measurements are used, e.g., via photon-counting detectors (5, 7) or threshold detectors (54), modeling measurement outcomes become extremely challenging even for supercomputers. It has been shown that, under standard complexity assumptions, sampling from the resulting probability distribution cannot be done in polynomial time using classical resources (2, 7).

For a Gaussian state with zero displacement and covariance matrix σ , the GBS distribution obtained by measuring the state with photon-counting detectors is given by (7)

$$p(S) = \frac{\text{Haf}(\mathcal{A}_S)}{n_1! \dots n_M! \sqrt{\det(\sigma + 1/2)}} \quad (4)$$

where $\mathcal{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} [1 - (\sigma + 1/2)^{-1}]$, and \mathcal{A}_S is a $2N \times 2N$ submatrix of \mathcal{A} , with $N = \sum_{j=1}^M n_j$. The set $S = (n_1, \dots, n_M)$ defines a measurement outcome, where n_j is the number of photons in mode j , and the submatrix \mathcal{A}_S is obtained by selecting rows and columns of \mathcal{A} , as described in (7). The function $\text{Haf}(\mathcal{A}_S)$ is the Hafnian of \mathcal{A}_S , a matrix function that is #P-Hard to approximate for worst-case instances (57, 58). For a $2N \times 2N$ matrix A , it is defined as

$$\text{Haf}(A) = \sum_{M \in \text{PMP}} \prod_{(i,j) \in M} A_{ij} \quad (5)$$

where PMP is the set of perfect matching permutations, namely, the possible ways of partitioning the set $1, \dots, 2N$ into subsets of size 2. When threshold detectors are used (54), the output is a binary variable s_j for each mode: $s_j = 1$ corresponds to a “click” from the j th detector that occurs whenever $n_j > 0$; on the other hand, $s_j = 0$ for $n_j = 0$. The probability distribution with threshold detectors can be obtained by summing infinitely many probabilities from Eq. 4 or via closed-form expressions that require evaluating an exponential number of matrix determinants (54).

GBS to find dense subgraphs

When A is the adjacency matrix of an unweighted graph G , the Hafnian of A is equal to the number of perfect matchings in G . Using mathematical properties of the Hafnian, it was shown in (15) that a GBS device can be programmed to sample from a distribution $p(S) \propto \frac{|\text{Haf}(\mathcal{A}_S)|^2}{c^N}$. The parameter c depends on the spectral properties of A and can be tuned to lower the probability of observing photon collisions, i.e., $n_j \geq 2$ for some j . More details are provided in section S1. In the collision-free subspace, \mathcal{A}_S is the adjacency matrix of the subgraph specified by the vertices j for which $n_j = 1$, and $\text{Haf}(\mathcal{A}_S)$ is equal to the number of perfect matchings in this subgraph. Therefore, a GBS device can be programmed to sample large-Hafnian subgraphs with high probability.

The density of a graph G is defined as the number of edges in G divided by the number of edges of the complete graph. Intuitively, a subgraph with a high number of perfect matchings should have a large density, a connection that was made rigorous in (59). This fact was used in (13) to show that GBS devices can be programmed to sample dense subgraphs with high probability. Hybrid quantum-classical optimization algorithms can be built by combining GBS random sampling with stochastic optimization algorithms for dense subgraph identification.

SUPPLEMENTARY MATERIALS

Supplementary material for this article is available at <http://advances.sciencemag.org/cgi/content/full/6/23/eaax1950/DC1>

REFERENCES AND NOTES

1. R. P. Feynman, Simulating physics with computers. *Int. J. Theor. Phys.* **21**, 467–488 (1982).
2. S. Aaronson, A. Arkhipov, The computational complexity of linear optics, in *Proceedings of The Forty-Third Annual ACM Symposium on Theory of Computing* (ACM, 2011), pp. 333–342.
3. P. Clifford, R. Clifford, The classical complexity of boson sampling, in *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms* (Society for Industrial and Applied Mathematics, 2018), pp. 146–155.
4. A. Neville, C. Sparrow, R. Clifford, E. Johnston, P. M. Birchall, A. Montanaro, A. Laing, Classical boson sampling algorithms with superior performance to near-term experiments. *Nat. Phys.* **13**, 1153–1157 (2017).
5. A. P. Lund, A. Laing, S. Rahimi-Keshari, T. Rudolph, J. L. O'Brien, T. C. Ralph, Boson sampling from a Gaussian state. *Phys. Rev. Lett.* **113**, 100502 (2014).
6. M. Bentivegna, N. Spagnolo, C. Vitelli, F. Flamini, N. Viggianiello, L. Latmiral, P. Mataloni, D. J. Brod, E. F. Galvão, A. Crespi, R. Ramponi, R. Osellame, F. Sciarrino, Experimental scattershot Boson sampling. *Sci. Adv.* **1**, e1400255 (2015).
7. C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, I. Jex, Gaussian Boson sampling. *Phys. Rev. Lett.* **119**, 170501 (2017).
8. Z. Vernon, N. Quesada, M. Liscidini, B. Morrison, M. Menotti, K. Tan, J. E. Sipe, Scalable squeezed light source for continuous variable quantum sampling. *Phys. Rev. Appl.* **12**, 064024 (2018).
9. S. Wolfram, Undecidability and intractability in theoretical physics. *Phys. Rev. Lett.* **54**, 735–738 (1985).
10. J. Huh, G. G. Guerreschi, B. Peropadre, J. R. McClean, A. Aspuru-Guzik, Boson sampling for molecular vibronic spectra. *Nat. Photon.* **9**, 615–620 (2015).
11. W. R. Clements, J. J. Renema, A. Eckstein, A. A. Valido, A. Lita, T. Gerrits, S. W. Nam, W. S. Kolthammer, J. Huh, I. A. Walmsley, Approximating vibronic spectroscopy with imperfect quantum optics. *J. Phys. B At. Mol. Phys.* **51**, 245503 (2017).
12. C. Sparrow, E. Martín-López, N. Maraviglia, A. Neville, C. Harrold, J. Carolan, Y. N. Joglekar, T. Hashimoto, N. Matsuda, J. L. O'Brien, D. P. Tew, A. Laing, Simulating the vibrational quantum dynamics of molecules using photonics. *Nature* **557**, 660–667 (2018).
13. J. M. Arrazola, T. R. Bromley, Using Gaussian Boson sampling to find dense subgraphs. *Phys. Rev. Lett.* **121**, 030503 (2018).
14. J. M. Arrazola, T. R. Bromley, P. Rebentrost, Quantum approximate optimization with Gaussian Boson sampling. *Phys. Rev. A* **98**, 012322 (2018).
15. K. Brádler, P.-L. Dallaire-Demers, P. Rebentrost, D. Su, C. Weedbrook, Gaussian boson sampling for perfect matchings of arbitrary graphs. *Phys. Rev. A* **98**, 032310 (2018).
16. D. B. Kitchen, H. Decornez, J. R. Furr, J. Bajorath, Docking and scoring in virtual screening for drug discovery: Methods and applications. *Nat. Rev. Drug Discov.* **3**, 935–949 (2004).
17. X.-Y. Meng, H.-X. Zhang, M. Mezei, M. Cui, Molecular docking: A powerful approach for structure-based drug discovery. *Curr. Comput. Aided Drug Des.* **7**, 146–157 (2011).
18. R. L. DesJarlais, R. P. Sheridan, J. S. Dixon, I. D. Kuntz, R. Venkataraghavan, Docking flexible ligands to macromolecular receptors by molecular shape. *J. Med. Chem.* **29**, 2149–2153 (1986).
19. B. K. Shoichet, I. D. Kuntz, D. L. Bodian, Molecular docking using shape descriptors. *J. Comput. Chem.* **13**, 380–397 (1992).
20. B. K. Shoichet, I. D. Kuntz, Matching chemistry and shape in molecular docking. *Protein Eng.* **6**, 723–732 (1993).
21. R. Dias, W. F. J. de Azevedo, Molecular docking algorithms. *Curr. Drug Targets* **9**, 1040–1047 (2008).
22. H. Alonso, A. A. Bliznyuk, J. E. Gready, Combining docking and molecular dynamic simulations in drug design. *Med. Res. Rev.* **26**, 531–568 (2006).
23. B. K. Shoichet, Virtual screening of chemical libraries. *Nature* **432**, 862–865 (2004).
24. M. Hernandez, M. Aramon, Enhancing quantum annealing performance for the molecular similarity problem. *Quant. Inf. Proc.* **16**, 133 (2017).
25. M. Hernandez, G. L. Gan, K. Linvill, C. Dukatz, J. Feng, G. Bhisetti, A quantum-inspired method for three-dimensional ligand-based virtual screening. *J. Chem. Inf. Model.* **59**, 4475–4485 (2019).
26. F. S. Kuhl, G. M. Crippen, D. K. Friesen, A combinatorial algorithm for calculating ligand binding. *J. Comput. Chem.* **5**, 24–34 (1984).
27. I. D. Kuntz, J. M. Blaney, S. J. Oatley, R. Langridge, T. E. Ferrin, A geometric approach to macromolecule-ligand interactions. *J. Mol. Biol.* **161**, 269–288 (1982).
28. I. Halperin, B. Ma, H. Wolfson, R. Nussinov, Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins* **47**, 409–443 (2002).
29. G. L. Warren, C. W. Andrews, A.-M. Capelli, B. Clarke, J. L. Londe, M. H. Lambert, M. Lindvall, N. Nevins, S. F. Semus, S. Senger, G. Tedesco, I. D. Wall, J. M. Woolven, C. E. Peishoff, M. S. Head, A critical assessment of docking programs and scoring functions. *J. Med. Chem.* **49**, 5912–5931 (2006).
30. B. G. Rao, U. K. Bandarage, T. Wang, J. H. Come, E. Perola, Y. Wei, S.-K. Tian, J. O. Saunders, Novel thiol-based TACE inhibitors: Rational design, synthesis, and SAR of thiol-containing aryl sulfonamides. *Bioorg. Med. Chem. Lett.* **17**, 2250–2253 (2007).
31. M. L. Moss, L. Sklair-Tavron, R. Nudelman, Drug Insight: Tumor necrosis factor-converting enzyme as a pharmaceutical target for rheumatoid arthritis. *Nat. Clin. Pract. Rheumatol.* **4**, 300–309 (2008).

32. S. Kalyanamoorthy, Y.-P. P. Chen, Structure-based drug design to augment hit discovery. *Drug Discov. Today* **16**, 831–839 (2011).
33. S.-Y. Yue, Distance-constrained molecular docking by simulated annealing. *Protein Eng.* **4**, 177–184 (1990).
34. O. Trott, A. J. Olson, AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.* **31**, 455–461 (2009).
35. M. Rarey, B. Kramer, T. Lengauer, G. Klebe, A fast flexible docking method using an incremental construction algorithm. *J. Mol. Biol.* **261**, 470–489 (1996).
36. M. D. Miller, S. K. Kearsley, D. J. Underwood, R. P. Sheridan, FLOG: A system to select 'quasi-flexible' ligands complementary to a receptor of known three-dimensional structure. *J. Comput. Aided Mol. Des.* **8**, 153–174 (1994).
37. T. J. A. Ewing, I. D. Kuntz, Critical evaluation of search algorithms for automated molecular docking and database screening. *J. Comput. Chem.* **18**, 1175–1189 (1997).
38. M. D. Miller, R. P. Sheridan, S. K. Kearsley, SQ: A program for rapidly producing pharmacophorically relevant molecular superpositions. *J. Med. Chem.* **42**, 1505–1514 (1999).
39. S.-Y. Yang, Pharmacophore modeling and applications in drug discovery: Challenges and recent advances. *Drug Discov. Today* **15**, 444–450 (2010).
40. A. M. Poole, R. Ranganathan, Knowledge-based potentials in protein design. *Curr. Opin. Struct. Biol.* **16**, 508–513 (2006).
41. J. Mintseris, B. Pierce, K. Wiehe, R. Anderson, R. Chen, Z. Weng, Integrating statistical pair potentials into protein complex prediction. *Proteins* **69**, 511–520 (2007).
42. H. Gohlke, G. Klebe, Statistical potentials and scoring functions applied to protein–ligand binding. *Curr. Opin. Struct. Biol.* **11**, 231–235 (2001).
43. R. M. Karp, Reducibility among Combinatorial Problems, in *Complexity of Computer Computations* (Springer, 1972), pp. 85–103.
44. Q. Wu, J.-K. Hao, A review on algorithms for maximum clique problems. *Eur. J. Oper. Res.* **242**, 693–709 (2015).
45. F. R. Chung, F. C. Graham, *Spectral Graph Theory* (American Mathematical Society, 1997).
46. I. Gutman, M. Ghorbani, Some properties of the Narumi–Katayama index. *Appl. Math. Lett.* **25**, 1435–1438 (2012).
47. W. Pullan, H. H. Hoos, Dynamic local search for the maximum clique problem. *J. Artif. Intell. Res.* **25**, 159–185 (2006).
48. W. Pullan, Phased local search for the maximum clique problem. *J. Comb. Optim.* **12**, 303–323 (2006).
49. G. Landrum, RDKit: Cheminformatics and machine learning software (2006). RDKit: Open-Source Cheminformatics Software; <http://rdkit.org/>.
50. R. Wang, X. Fang, Y. Lu, S. Wang, The PDBbind database: Collection of binding affinities for protein–ligand complexes with known three-dimensional structures. *J. Med. Chem.* **47**, 2977–2980 (2004).
51. H. Gohlke, M. Hendlich, G. Klebe, Knowledge-based scoring function to predict protein–ligand interactions. *J. Mol. Biol.* **295**, 337–356 (2000).
52. S. A. Hindle, M. Rarey, C. Buning, T. Lengauer, Flexible docking under pharmacophore type constraints. *J. Comput. Aided Mol. Des.* **16**, 129–149 (2002).
53. M. L. Verdonk, V. Berdini, M. J. Hartshorn, W. T. M. Mooij, C. W. Murray, R. D. Taylor, P. Watson, Virtual screening using protein–ligand docking: Avoiding artificial enrichment. *J. Chem. Inf. Comput. Sci.* **44**, 793–806 (2004).
54. N. Quesada, J. M. Arrazola, N. Killoran, Gaussian Boson sampling using threshold detectors. *Phys. Rev. A* **98**, 062322 (2018).
55. C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, S. Lloyd, Gaussian quantum information. *Rev. Mod. Phys.* **84**, 621–669 (2012).
56. S. D. Bartlett, B. C. Sanders, S. L. Braunstein, K. Nemoto, Efficient classical simulation of continuous variable quantum information processes. *Phys. Rev. Lett.* **88**, 097904 (2002).
57. E. R. Caianiello, On quantum field theory—I: Explicit solution of Dyson's equation in electrodynamics without use of feynman graphs. *Il Nuovo Cimento (1943-1954)* **10**, 1634–1652 (1953).
58. A. Björklund, B. Gupt, N. Quesada, A faster hafnian formula for complex matrices and its benchmarking on a supercomputer. *J. Exp. Algorithmics* **24**, 1 (2018).
59. M. Aghabali, S. Akbari, S. Friedland, K. Markström, Z. Tajfirouz, Upper bounds on the number of perfect matchings and directed 2-factors in graphs with given number of vertices and edges. *Eur. J. Combinatorics* **45**, 132–144 (2015).
60. B. Gupt, J. M. Arrazola, N. Quesada, T. R. Bromley, Classical benchmarking of Gaussian Boson Sampling on the Titan supercomputer. arXiv:1810.00900 [quant-ph] (1 October 2018).
61. Z. Liu, Y. Li, L. Han, J. Li, J. Liu, Z. Zhao, W. Nie, Y. Liu, R. Wang, PDB-wide collection of binding data: Current status of the PDBbind database. *Bioinformatics* **31**, 405–412 (2014).
62. R. Wang, X. Fang, Y. Lu, C.-Y. Yang, S. Wang, The PDBbind database: Methodologies and updates. *J. Med. Chem.* **48**, 4111–4119 (2005).
63. H. Gohlke, M. Hendlich, G. Klebe, Predicting binding modes, binding affinities and 'hot spots' for protein–ligand complexes using a knowledge-based scoring function. *Perspect. Drug Discov. Design* **20**, 115–144 (2000).
64. W. T. M. Mooij, M. L. Verdonk, General and targeted statistical potentials for protein–ligand interactions. *Proteins* **61**, 272–287 (2005).
65. H.-S. Zhong, L.-C. Peng, Y. Li, Y. Hu, W. Li, J. Qin, D. Wu, W. Zhang, H. Li, L. Zhang, Z. Wang, L. You, X. Jiang, L. Li, N.-L. Liu, J. P. Dowling, C.-Y. Lu, J.-W. Pan, Experimental Gaussian Boson sampling. *Sci. Bull.* **64**, 511–515 (2019).
66. B. Bollobás, P. Erdős, Cliques in random graphs, in *Mathematical Proceedings of the Cambridge Philosophical Society* (Cambridge Univ. Press, 1976), vol. 80, pp. 419–427.
67. L. Banchi, S. L. Braunstein, S. Pirandola, Quantum fidelity for arbitrary Gaussian States. *Phys. Rev. Lett.* **115**, 260501 (2015).

Acknowledgments: We thank T. R. Bromley, N. Killoran, N. Quesada, C. Weedbrook, and M. Schuld for fruitful discussions. L.B. is currently employed by the University of Florence, Italy. All the work leading to this manuscript has been carried out while L.B. was working full time for Xanadu. **Funding:** The authors acknowledge that they received no funding in support of this research. **Author contributions:** M.F., T.B., and C.I. formulated the method mapping molecular docking to maximum weighted clique and generated the binding interaction graphs. L.B. and J.M.A. devised the GBS algorithm for maximum weighted clique and performed the various analyses and simulations for testing the algorithm. All authors contributed to the writing of the paper. **Competing interests:** J.M.A. is employed and holds stock options at a quantum computing startup developing the photonic technology presented in this paper. T.B., M.F., and C.I. are cofounders, employees, and shareholders of ProteinQure Inc., which is a startup working on computational drug design. The authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper and/or the Supplementary Materials. Additional data related to this paper may be requested from the authors.

Submitted 1 March 2019

Accepted 3 April 2020

Published 5 June 2020

10.1126/sciadv.aax1950

Citation: L. Banchi, M. Fingerhuth, T. Babej, C. Ing, J. M. Arrazola, Molecular docking with Gaussian Boson Sampling. *Sci. Adv.* **6**, eaax1950 (2020).

Molecular docking with Gaussian Boson Sampling

Leonardo Banchi, Mark Fingerhuth, Tomas Babej, Christopher Ing and Juan Miguel Arrazola

Sci Adv **6** (23), eaax1950.
DOI: 10.1126/sciadv.aax1950

ARTICLE TOOLS

<http://advances.sciencemag.org/content/6/23/eaax1950>

SUPPLEMENTARY MATERIALS

<http://advances.sciencemag.org/content/suppl/2020/06/01/6.23.eaax1950.DC1>

REFERENCES

This article cites 60 articles, 1 of which you can access for free
<http://advances.sciencemag.org/content/6/23/eaax1950#BIBL>

PERMISSIONS

<http://www.sciencemag.org/help/reprints-and-permissions>

Use of this article is subject to the [Terms of Service](#)

Science Advances (ISSN 2375-2548) is published by the American Association for the Advancement of Science, 1200 New York Avenue NW, Washington, DC 20005. The title *Science Advances* is a registered trademark of AAAS.

Copyright © 2020 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works. Distributed under a Creative Commons Attribution License 4.0 (CC BY).