Automatic design of functional molecules and materials

Koji Tsuda
University of Tokyo / NIMS / RIKEN AIP
Automatic Materials Design

Machine Learning

Simulation (DFT etc)

Experiments

Experimental Design

Data
Materials Design Examples

• Bayesian Optimization
  – Design of Si-Ge nanostructures (Phys Rev X 2017)

• Monte Carlo Tree Search (MCTS)
  – ChemTS: Design of organic compounds by deep learning and tree search (STAM 2017)
Screening by first principles calculations alone

First Principles Calc.

<table>
<thead>
<tr>
<th>Mat. 1</th>
<th>Mat. 2</th>
<th>Mat. 3</th>
<th>Mat. 4</th>
<th>Mat. 5</th>
<th>Mat. 6</th>
<th>Mat. 7</th>
<th>Mat. 8</th>
<th>Mat. 9</th>
<th>Mat. 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score 1</td>
<td>Score 2</td>
<td>Score 3</td>
<td>Score 4</td>
<td>Score 5</td>
<td>Score 6</td>
<td>Score 7</td>
<td>Score 8</td>
<td>Score 9</td>
<td>Score 10</td>
</tr>
</tbody>
</table>
Bayesian Optimization
(Jones et al., 1998)

• Find best data points with minimum number of observations

• Choose next point to observe to discover the best ones as early as possible
Bayesian Optimization (1)

First Principles Calc.

Score 1  Score 2  Score 3
Bayesian Optimization (2)

First Principles Calc.

Predicted Scores

Predicted Variances
Bayesian Optimization (3)

<table>
<thead>
<tr>
<th>Mat. 1</th>
<th>Mat. 2</th>
<th>Mat. 3</th>
<th>Mat. 8</th>
<th>Mat. 4</th>
<th>Mat. 5</th>
<th>Mat. 6</th>
<th>Mat. 7</th>
<th>Mat. 9</th>
<th>Mat. 10</th>
</tr>
</thead>
</table>

First Principles Calc.

<table>
<thead>
<tr>
<th>Score 1</th>
<th>Score 2</th>
<th>Score 3</th>
<th>Score 8</th>
</tr>
</thead>
</table>

8
Bayesian Optimization (4)

<table>
<thead>
<tr>
<th>Mat. 1</th>
<th>Mat. 2</th>
<th>Mat. 3</th>
<th>Mat. 8</th>
<th>Mat. 4</th>
<th>Mat. 5</th>
<th>Mat. 6</th>
<th>Mat. 7</th>
<th>Mat. 9</th>
<th>Mat. 10</th>
</tr>
</thead>
</table>

First Principles Calc.

<table>
<thead>
<tr>
<th>Score 1</th>
<th>Score 2</th>
<th>Score 3</th>
<th>Score 8</th>
<th>Pred. Score 4</th>
<th>Pred. Score 5</th>
<th>Pred. Score 6</th>
<th>Pred. Score 7</th>
<th>Pred. Score 9</th>
<th>Pred. Score 10</th>
</tr>
</thead>
</table>

| Var. 4 | Var. 5 | Var. 6 | Var. 7 | Var. 9 | Var. 10 |
Where to observe next?
Gaussian Process

Current Maximum

Measured Value

Explanatory Variable
Maximum probability of improvement

Current Maximum

Measured Value

Explanatory Variable
**Question:** How to organize 16 alloy atoms (Si: 8, Ge: 8) to obtain the largest and smallest interfacial thermal conductance?

**Descriptors:** $C_8^8 = 12,870$

<table>
<thead>
<tr>
<th>Case</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Calculator:** Atomistic Green’s Function (AGF): Phonon transmission

**Evaluator:** Interfacial Thermal Conductance (ITC)

**Optimization method:** Thompson Sampling (Bayesian Optimization)
Optimal structures were obtained by calculating only 3.4% of all candidates.
**Topic:** Arrange 10-layer superlattices structure (5 layers Si + 5 layers of Ge) between Si and Si to obtain minimal thermal conductance (1 layer thickness = 5.43 Å)

**Descriptors:**

\[ C_{10}^5 = 252 \]

<table>
<thead>
<tr>
<th>Case</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Best Structure:** (1101010001)
Mastering the game of Go with deep neural networks and tree search

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. Here we introduce a new approach to computer Go that uses ‘value networks’ to evaluate board positions and ‘policy networks’ to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.
Atom assignment problem

**Question:** How to organize 16 alloy atoms (Si: 8, Ge: 8) to obtain the largest and smallest interfacial thermal conductance?

**Discrete black-box optimization:**

Maximize \( f(x) \), \( x \in \{0, 1, k - 1\}^N \)

subject to \( \sum_{\ell=1}^N I(x_\ell = j) = n_j, \ j = 0, \ldots, k - 1 \)
Search space is too large for Bayesian Optimization (Jones et al., 1998)

- Need to compute scores for all candidates
Monte Carlo Tree Search

• Candidates at leafs of search tree
• Merit at leafs: Thermal Conductivity
• Merit at intermediate node by Rollout
  – Random traverse to a leaf (engineered)
  – UCB score: average merit + Premium
MCTS steps

• Initially, only root node exists
• j-th level corresponds to $x_j$
• Tree grows by iterating the following 4 steps

Selection

Expansion

Simulation

Backpropagation

0110110010111000
0111010101001010
0111010101001010
0111010101001010
Experiments: Design time

• Total time = Design time + Simulation time
# calculations vs success rate

- Success rate = Fraction of discovering the best in 100 runs
# calculations vs success rate

- Success rate = Fraction of discovering the best in 100 runs
Total time vs achieved TC (N=22): fast design = more calculations
MCTS Summary

• For large problems, BO is too slow
• MCTS is a viable alternative to BO
• Unlike GA, constraints are easily incorporated

• MCTS does not require descriptors
• ML-techniques can be combined with MCTS (e.g., alpha GO)
de novo Molecular Generation

Machine learning (prediction)

Generation of molecules with desired properties

Properties:
- Binding Score
- Bio Activity
- ADMET
- ...

Chemical structure of a molecule with functional groups.
SMILES

Cc3ccc(c2nc(CCCCO/N=C(CCC(O)=O)c1ccccc1)c(C)o2)cc3

Atom: {C, c, o, O, N, F, [C@@H], n, -, S, Cl, [O-], [C@H], [NH+], [C@], s, Br, [nH], [NH3+], [NH2+], [C@@], [N+], [nH+], [S@], [N-], [n+], [S@@], [S-], I, [n-], P, [OH+], [NH-], [P@@H], [P@@], [PH2], [P@], [P+], [S+], [o+], [CH2-], [CH-], [SH+], [O+], [s+], [PH+], [PH], [S@@+]}
Bonds: {/, =, ¥#}
Ring: {1, 2, 3, 4, 5, 6, 7, 8, 9}
Branch: {(), ()}
De novo Molecular Generation

• Most existing methods make molecules by combining predetermined fragments

• De novo generation by deep neural networks
  – Variational autoencoder (Kusner et al., ICML 2017)
  – Recurrent neural network (Segler et al., Arxiv, 2017)

  – MCTS + Recurrent Neural Network
Generation with VAE

• Variational Auto Encoder (VAE).

• Problem
  – Low chance of creating valid SMILES (0.4%~5%)
  – Slow

ChemTS: MCTS and RNN

- Search tree of SMILES
- Grow tree by MCTS
- Intermediate node = prefix of SMILES
- Rollout: String completion by RNN

High chance of valid SMILES creation (>40%) Fast generation
RNN: trained by 250,000 compounds

- **Input**: String $s_1, \ldots, s_T$
- **Output**: Distribution of string $P(y_1), \ldots, P(y_T)$
- **Output string is right-shift of input string**

![Diagram](attachment:image.png)
MCTS

(a) Selection

(b) Expansion

(c) Simulation

(d) Backpropagation

UCB Score

\[ u_i = \frac{w_i}{v_i} + C \sqrt{\frac{2 \ln V_{parent}}{v_i}} \]

Evaluate drug-likeliness score

\[ J(m) = \log P(m) - SA(m) - RingPenalty(m) \]
Table 1. Maximum score \( J \) at time points 2, 4, 6 and 8 hours achieved by different molecular generation methods. The rightmost column shows the number of generated molecules per minute. The average values and standard deviations over 10 trials are shown.

<table>
<thead>
<tr>
<th>Method</th>
<th>2h</th>
<th>4h</th>
<th>6h</th>
<th>8h</th>
<th>Molecules/Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChemTS</td>
<td>4.91 ( \pm ) 0.38</td>
<td>5.41 ( \pm ) 0.51</td>
<td>5.49 ( \pm ) 0.44</td>
<td>5.58 ( \pm ) 0.50</td>
<td>40.89 ( \pm ) 1.57</td>
</tr>
<tr>
<td>RNN+BO</td>
<td>3.54 ( \pm ) 0.27</td>
<td>4.46 ( \pm ) 0.24</td>
<td>4.46 ( \pm ) 0.24</td>
<td>4.46 ( \pm ) 0.24</td>
<td>8.33 ( \pm ) 0.00</td>
</tr>
<tr>
<td>Only RNN</td>
<td>4.51 ( \pm ) 0.27</td>
<td>4.62 ( \pm ) 0.26</td>
<td>4.79 ( \pm ) 0.25</td>
<td>4.79 ( \pm ) 0.25</td>
<td>41.33 ( \pm ) 1.42</td>
</tr>
<tr>
<td>CVAE+BO</td>
<td>-30.18 ( \pm ) 26.91</td>
<td>-1.39 ( \pm ) 2.24</td>
<td>-0.61 ( \pm ) 1.08</td>
<td>-0.006 ( \pm ) 0.92</td>
<td>0.14 ( \pm ) 0.08</td>
</tr>
<tr>
<td>GVAE+BO</td>
<td>-4.34 ( \pm ) 3.14</td>
<td>-1.29 ( \pm ) 1.67</td>
<td>-0.17 ( \pm ) 0.96</td>
<td>0.25 ( \pm ) 1.31</td>
<td>1.38 ( \pm ) 0.91</td>
</tr>
<tr>
<td>SMILES representation</td>
<td>$J$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------------------------------------------------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2NCce2ccc(Cl)cc2Cl)c2ecce2c1OC(F)F)c1ecce2ecce12</td>
<td>6.56</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2NCce2ccc(Cl)cc2Cl)c1ec1C1=CCCCCC1)c1cc(F)cc(Cl)c1</td>
<td>6.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2N=C(SC2CCCCC2)c2ecce2)cc(Cl)c1Cl)c1cc2ecce2ecce1n1</td>
<td>6.34</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Oc2ec(Cl)cc2Cl)cce1Nc1cc(Cl)cc1Cl)c1ecce(Cl)cc1</td>
<td>6.33</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2Cl)c(Cl)cc1Br)N(c1ecce1)c1ecce(Cl)cc1</td>
<td>6.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Oc2ec(Cl)ccce2Oc2ec(-c3ecce3)ec2)cc1Cl)c1eccc1</td>
<td>6.19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2Cl)c(Cl)c(C(=O)N(Cc2ecce2)c2ecce2)c1Cl)c1ecce1F</td>
<td>6.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Oc2ec(Cl)cc2Cl)c(Cl)c1Cl)c1nec1-c1ecce(Sc2ecce2)cc1</td>
<td>6.007</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2NCc2ec(Cl)cc2Cl)c2ncec2e1Cl)c1cc(Cl)cc1</td>
<td>6.0067</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O=C(Nc1cc(Nc2e(Cl)ecce2NCc2ec(Cl)cc2Cl)c(Cl)cc1Cl)c1cc(F)cc1Cl</td>
<td>6.0062</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>