Supplemental Materials: Efficient two-dimensional defect-free dual-species atom arrays rearrangement algorithm with near-fewest atom moves*

Zhi-Jin Tao(陶知进),[1,2,†] Li-Geng Yu(余立庚),[1,2,†] Peng-Xu(许鹏),[1,3,‡] Jia-Yi Hou(侯嘉毅),[1] Xiao-Dong He(何晓东),[1] and Ming-Sheng Zhan(詹明生)[1,3]

[1] *State Key Laboratory of Magnetic Resonance and Atomic and Molecular Physics, Wuhan Institute of Physics and Mathematics, Innovation Academy for Precision Measurement Science and Technology, Chinese Academy of Sciences, Wuhan 430071, China*

[2] *School of Physics and Technology, Wuhan University, Wuhan 430072, China*

[3] *Wuhan Institute of Quantum Technology, Wuhan 430206, China*

I. DETAILED STEPS

The general algorithm structure of each step can be summarized as a loop structure (Fig.1). The loop body can be characterized as two steps: Firstly, get the list of objects needed to be processed, e.g., misplaced atoms, obstacle atoms, or empty sites, sorted by particular priority; Secondly, find ways to process the object at first of the list. If the object has no feasible way to handle, try the next one. Otherwise, delete the object in the list and try the next one. If the next one is none, determine whether the loop should be terminated. The loop terminates when the list is empty, or all the objects fail to be processed. Otherwise, continue the next loop. The following is a general explanation of each step:

The first step aims to move the misplaced atoms to the target empty sites or outside sites. We made the following considerations to improve the connectivity of the empty site graph during each move. First, get the list of misplaced atoms. Optionally, sort the list by atom partition number. Second, find the empty site s with partition number 0 which the atom can reach [Fig.2.(a,b)]. To approach the fewest atom moves, a misplaced atom is preferentially moved to a target empty site, and the target region mustn't form a new misplaced atom or a new blocked misplaced atom after each move [Fig.2.(c)]. Fig.2 demonstrates how step1 is carried out. In this example, all the misplaced atoms are removed. The connected components of the graph of the empty site have reached a lower level, which provides a good condition for the execution of the following steps.

Step 2 removes the misplaced atoms unsolved by step 1 and eliminates the enclosed target empty sites region. If one algorithm ignores these cases, the success rate will drop catastrophically. Before illustration of step 2, we introduce the first-layer obstacle atom and block number concept. The first layer obstacle atom blocking misplaced atoms are the atom whose adjacent empty sites or the atoms itself can be reached by the misplaced atoms. The block number is the number of misplaced atoms the obstacle atom blocks. The scheme of the first layer obstacle atom and block number is shown in Fig.4.

First, we get the list of first-layer obstacle atoms and sort them by block number. The first-layer obstacle atom with a greater block number is moved to a higher priority. Fig.3. (a) shows the first-layer obstacle atoms

* Supported by the National Key R&D Program of China (Grant No. 2021YFA1402001, No. 2017YFA0304501), the Youth Innovation Promotion Association CAS No. Y2021091 and No. 2019325, the National Natural Science Foundation of China under Grant No. U20A2074 and No. 12074391, the K.C. Wong Education Foundation (Grant No. GJTD-2019-15)

† Z.T. and L.Y contributed equally to this work.

‡ xupeng@apm.ac.cn

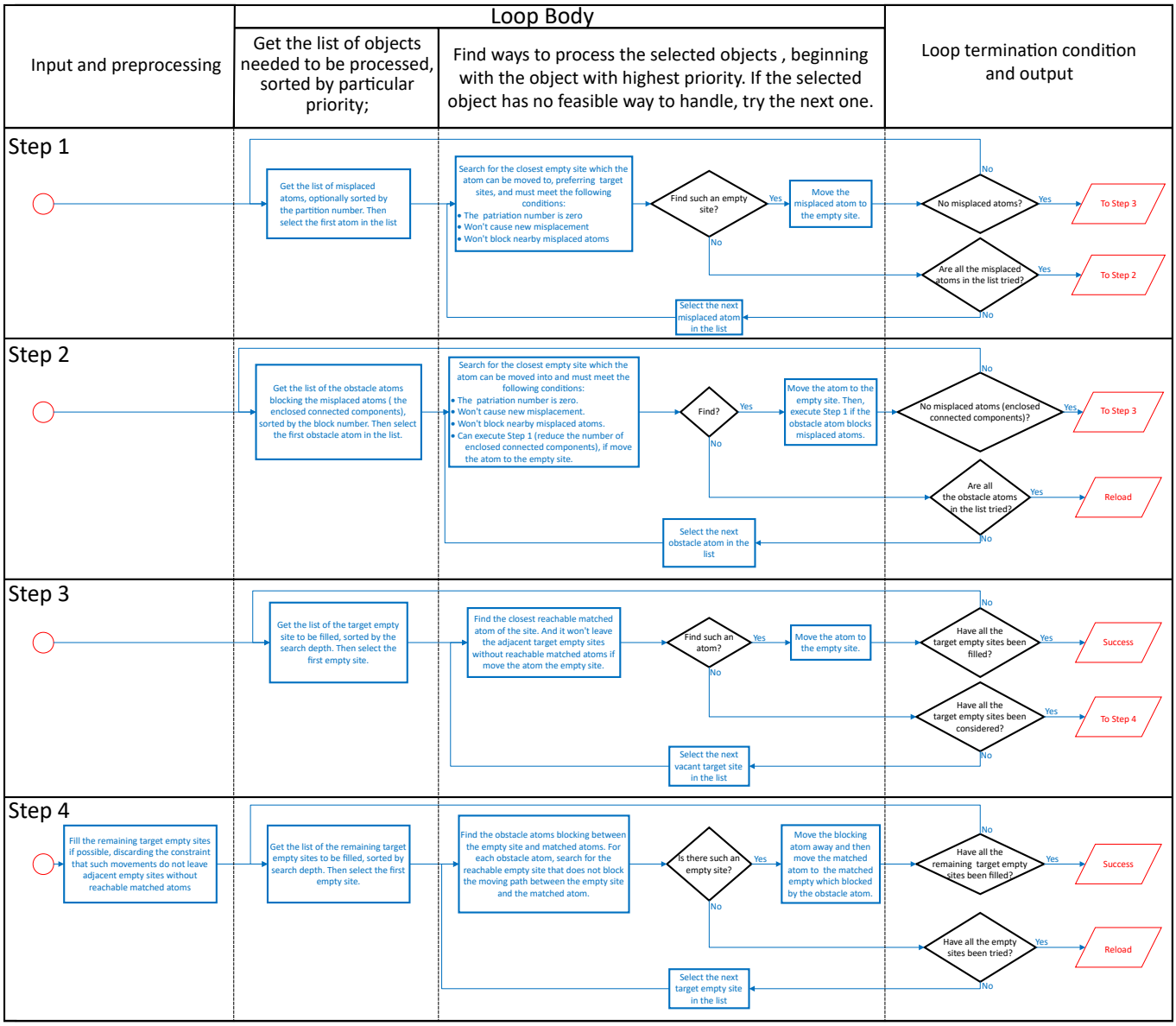

Fig.1 Detailed flowchart for each step

blocking misplaced atoms, labeled by their block number. Move the obstacle atom to the empty site that meets the following conditions: won't cause new misplacement or block adjacent misplaced atoms after the move. The partition number of the empty site is 0 when treating the atom to be moved as an empty site. If such an empty site exists, move the atom to the empty site, execute step 1, and process the next first-layer obstacle atom. Otherwise, process the next first-layer obstacle atom with lower priority. When all the misplaced atoms are moved to the matched empty sites, move the boundary atoms of the enclosed target empty sites region to the empty sites that satisfy the former three conditions until there is no enclosed target empty region. Fig.3. (b) demonstrates the moving atoms and move paths of step 2. The probability of executing step 2 is shown in Fig.3.(d). After the step 2, all the target empty sites regions are connected with the outside region, which brings convenience for the step 3.

Step 3 is to fill target empty sites. As target empty sites s generally have positive partition number

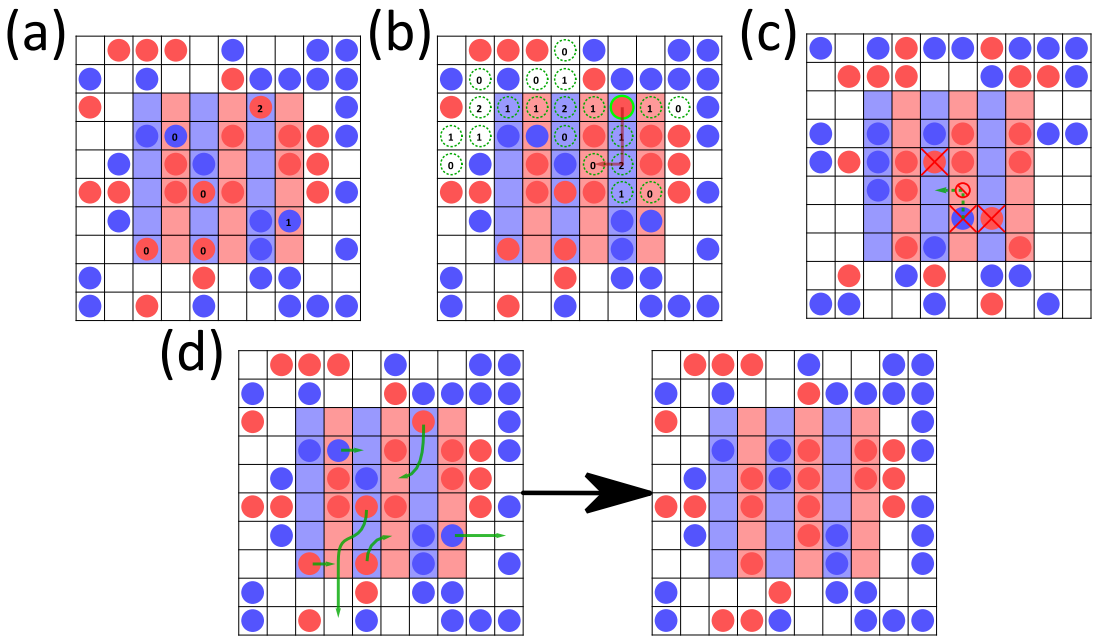

Fig.2 Schemes of step 1. (a) The initial atom array of the example. Misplaced atoms are labeled by their partition numbers. (b) The first move in executing step 1 for the initial atom array. The circled atom is selected to be moved firstly because it has the biggest partition number. Empty sites reached by the selected atom are marked by the dotted circle and labeled by their partition number. Step 1 decides to move the selected atom along the arrow, as the empty destination site is the closest matched empty site with zero partition number. (c) An example that a move will block the adjacent misplaced atoms. (d) Move paths of misplaced atoms in step 1 and the configuration of the atom array after executing step 1.

$P(G, s) > 0$ in this stage, we changed the heuristic strategy: preferentially fill the empty target site deeper to the boundary of the target region. The method to determine the depth of an empty site to the boundary of target region is shown in Fig.5.(a). A move in the filling process is to move the closest matched atom outside to the empty target site. To improve the success rate of step 3, such a move in the filling process shouldn't leave adjacent target sites having any reachable matched atoms in the outside region (An example is shown in Fig.5.(c)).

To solve the cases left unsolved by step 3, we introduced step 4. Fig.6.(b) shows the probability of executing step 4 decreases with the grid size increasing, demonstrating the former 3 steps are almost enough for the larger grid size. The preprocessing step of step 4 is filling the target empty sites by matched atoms in the outside region, ignoring the restriction that such a move will not leave adjacent target sites has no reachable matched atom in the outside region in step 3, and add a restriction that such a move will not form any enclosed empty sites in the target region (see Fig.6.(b)) Then find out the obstacle atoms blocking the shortest moving path of an atom in the outside region to the matched empty target site. Move one obstacle atom to empty sites not blocking the moving path if possible, and then fill the empty target site by the matched atom through the shortest moving path (see Fig.6.(c)).

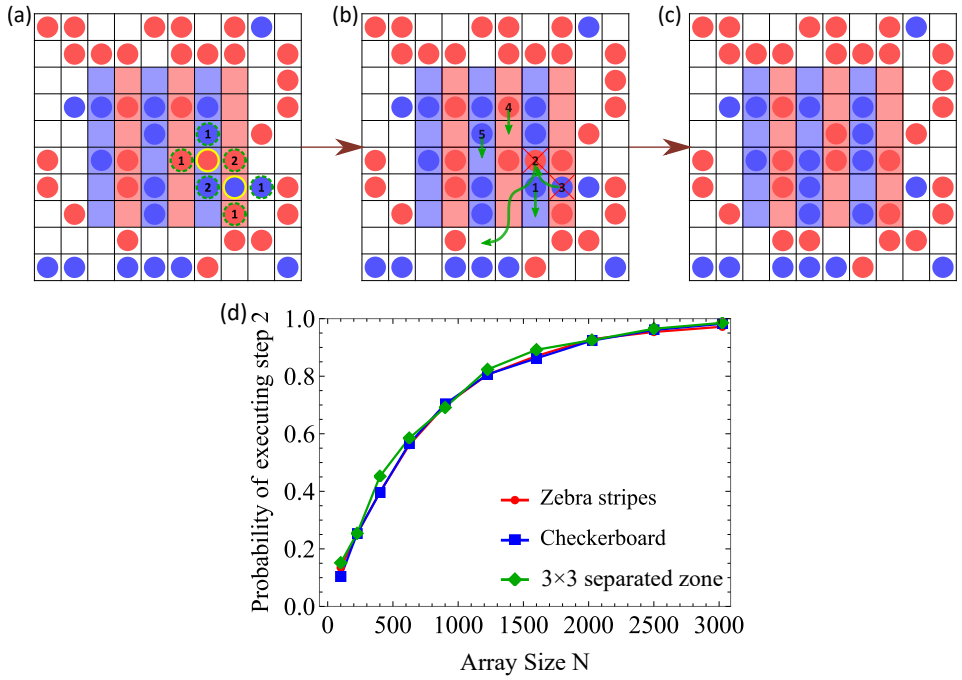

Fig.3 Scheme of step 2. (a) The dotted and circled atoms represent the obstacle atoms, labeled by their block numbers. The atoms circled with yellow lines are misplaced atoms. (b) Move paths of step 2 for this example. (c) The atom array configuration after executing step 2. (d) The probability of executing step 2.

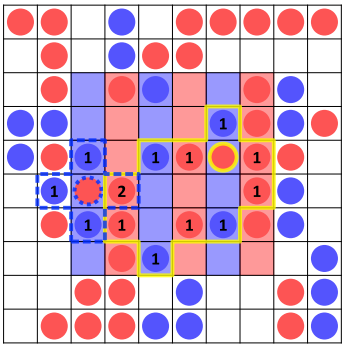

Fig.4 Scheme of the first layer obstacle atom and block number.The first layer obstacle atoms for each misplaced atom circled with solid (dashed) line are framed with solid (dashed) line. The misplaced atom circled by dashed line has no empty site to move into, while the misplaced atom circled by solid line has no empty site satisfying conditions required in step 1 to move into. The first layer obstacle atoms are labeled by their block number.

II. THE DEMONSTRATION PROGRAM

The demonstration program for HCOA is uploaded to Github:

https://github.com/YLG-WHU/Dual-species-atom-rearrangement-demonstration.

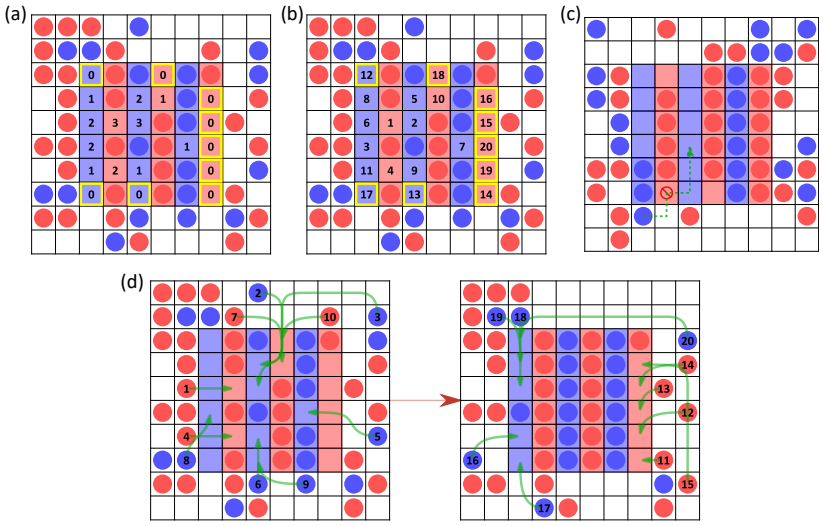

Fig.5 Scheme of step 3. (a) Method to determine depth to the boundary of target region. First, find the empty target sites with zero depth, which are neighbor to the boundary and their adjacent sites in outside region are not all unmatched atoms. The zero depth sites are framed. Second, do the broad first search, beginning with the zero depth sites until all empty target sites are labeled by their depth to the boundary of target region. (b) The priority level of empty target sites. The empty target site deeper to the boundary and having less neighbor empty sites is prior to be filled. (c) An example of such a move which leaves the adjacent target sites having no reachable matched atom in the outside region. (d) An example of executing step 3. All empty target sites are filled. The moved atoms are labeled by their move orders.

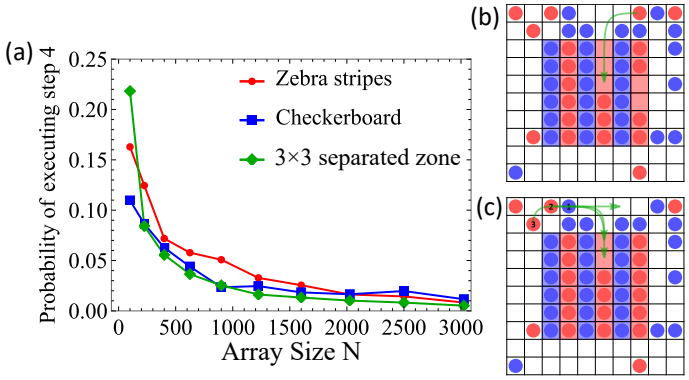

Fig.6 Scheme of step 4. (a) Probability of executing step 4 for three different target site patterns vs grid size. (b) The preprocessing step of step 4. Fill the remaining target empty sites if possible, discarding the constraint that such movements do not leave adjacent empty sites without reachable matched atoms. (c) An example of step 4: the obstacle atom labeled by 1 is moved away, and then the remaining target empty sites are filled by matched atoms labeled by 2 and 3

III. THE IDEAL ATOM MOVES

The ideal number of atom moves M_m is defined as the minimum number of atom moves needed to rearrange the initially loaded atoms to the target configuration if any atom can be directly moved to any site. After loading

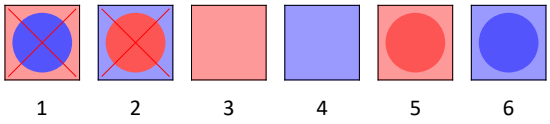

Fig.7 The six types of site-atom patterns

the atoms stochastically, there are six types of site-atom patterns [see in Fig.7]. The type-1 pattern and type-2 pattern refer to the sites filled with misplaced atoms. The type-3 pattern and type-4 pattern represent the empty sites. The type-5 pattern and type-6 pattern are the sites filled with the matched atoms. Then $N_m = max(N_1, N_2 + N_4) + max(N_2, N_1 + N_3)$, where $N_i, i = 1..6$ is the number of each pattern.

The target blue and red sites should be filled with the matched atoms, so we first move the misplaced atoms to such target sites. Obviously, the misplaced atoms are eliminated in the first step if $N_1 \leq N_2 + N_4$ and $N_2 \leq N_1 + N_3$. In this way, then move the outside atoms to the matched empty sites, and the minimum number of atom moves is $N_2 + N_4 + N_1 + N_3$. It should be noted that the type-2 site and type-1 site is allowed to be filled with the matched atom and the misplaced atom simultaneously if the misplaced blue atoms or red atoms are more than the sites not filled with the matched atoms, i.e., $N_1 > N_2 + N_4$ or $N_2 > N_1 + N_3$. In this case, the target sites are filled with the matched atoms, and some of the target sites are filled with two different atoms; then, move the extra misplaced atoms to the empty outside site, and the target configuration is achieved. So $N_m = max(N_1, N_2 + N_4) + max(N_2, N_1 + N_3)$.

IV. COMPARISON OF THE SUCCESS RATE BETWEEN HCOA AND HHA

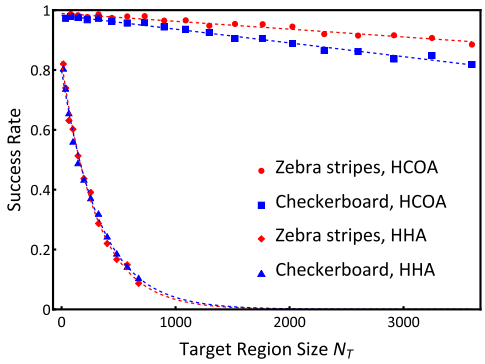

Fig.8 The comparison of success rate between HCOA and HHA. The success rate of HHA algorithm is fitted with an exponential decay function $y = a \times 10^{-bx}$, for Zebra strips pattern $a = 0.83 \pm 0.03$, $b = -0.00140 \pm 0.00004$ and for Checkerboard pattern $a = 0.80 \pm 0.01$, $b = -0.00130 \pm 0.00003$; The success rate of HCOA algorithm is fitted with a linear curve $y = a + bx$, for Zebra strips pattern $a = 0.989 \pm 0.003$, $b = (2.6 \pm 0.2) \times 10^{-5}$ and for Checkerboard pattern $a = 0.982 \pm 0.003$, $b = (4.6 \pm 0.1) \times 10^{-5}$.

Fig.8 shows the comparison of the success rate between HCOA and HHA. From the graph, HCOA shows a much higher success rate compared with HHA and remains at a high level with the target region size increasing. By taking the connectivity of the graph as a critical consideration at the main steps, the success rate of HCOA decreases linearly with the number of atoms increasing and remains larger than 0.80 as the number of target

atoms increase to 3600.

V. ANALYSIS FOR THE EFFECTIVENESS OF CONNECTIVITY

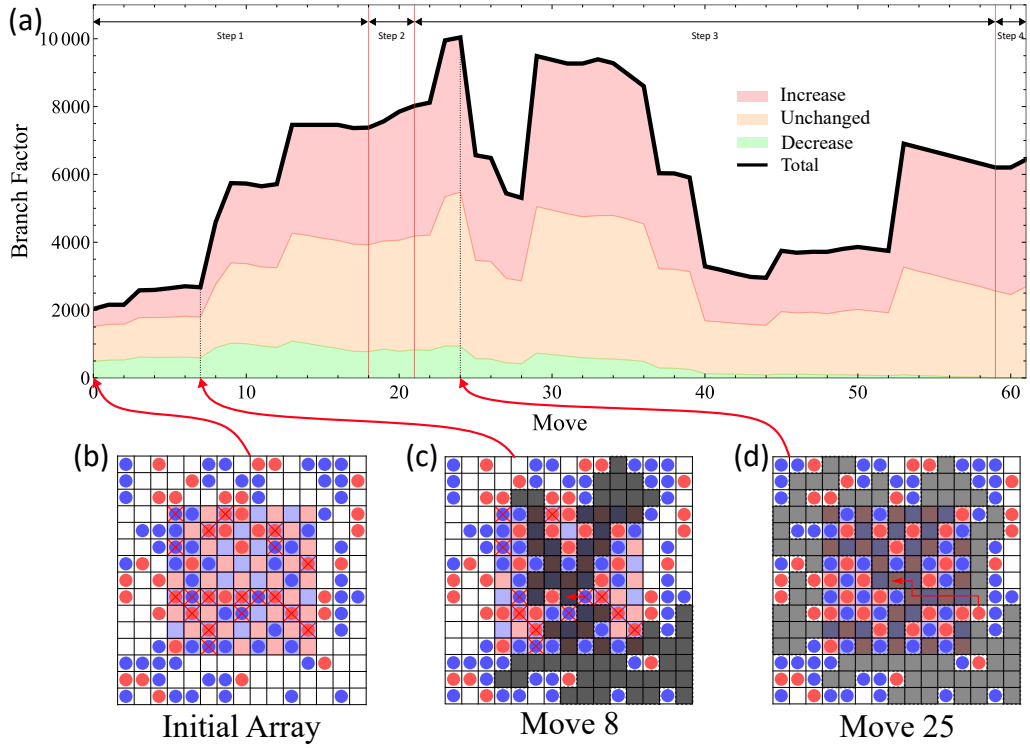

Fig.9 Branch factor vs moves for an example of 15 × 15 square atom array rearrangement. (a) Total possible atom move paths(branch factor) vs moves in one rearrangement process.The pink, orange, green regions represent the possible atom move paths which will increase, unchange, decrease the total atom configuration entropy respectively (C. Sheng *et al.*'s work[1] provides calculation formula for atom array configuration entropy). (b) The initial array configuration. (c) Atom array configuration in Move 8. The branch factor increases significantly due to the misplaced atom separating two large connected components (colored dark) being removed. (d) Atom array configuration in Move 25. The branch factor reaches maximum with the connected components number dropping a lot and thus connectivity of the graph improves a lot.

Fig.9 shows the branching factor(the total possible atom move paths) vs. rearrangement steps in our algorithm. Fig.9 suggests that the branching factor maximizes when the empty sites are nearly connected, which proves the importance of the graph's connectivity.

REFERENCES

[1] C. Sheng, J.-Y. Hou, X.-D. He, K.-P. Wang, R.-J.Guo, B. Mamat, P. Xu, M. Lin, J. Wang and M.-S. Zhan. 2022 *Physical Review Letters* **128** 083202