Towards assembly of arrays of 1000 atoms

Jean-Samuel Tettekpoe[1,2], Rene Villela[1], Yeelai Chew[1,3], Sylvain de Léséleuc[1], Takafumi Tomita[1], and Kenji Ohmori[1,3]

[1]Institute for Molecular Science, National Institutes of Natural Sciences, Japan. [2]École polytechnique, Palaiseau, France. [3]SOKENDAI (The Graduate University for Advanced Studies), Japan.

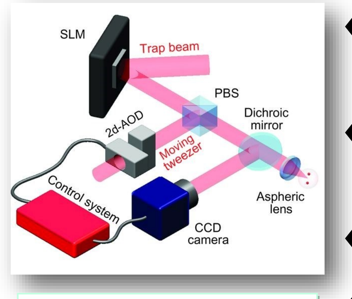
National Institutes of Natural Sciences Institute for Molecular Science

Goal 6 moonshot 大森 PM

大規模・高コヒーレンスな動的原子アレー型・誤り耐性量子コンピュータ

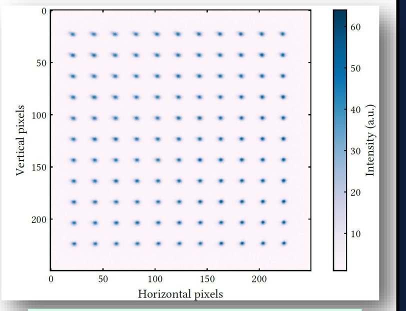
Introduction

Two-dimensional Rubidium arrays...



Atom trapping setup
Reproduced from [2]

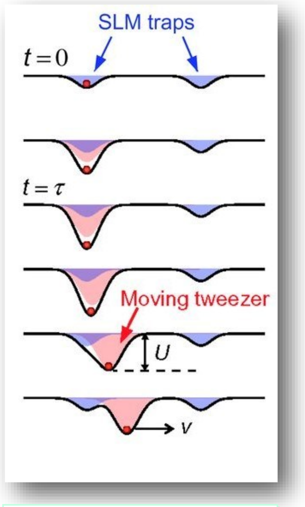
- SLM gives the best homogeneous tweezer arrays and freedom in array geometry [1]
- 50-60% chance of loading an atom into an SLM tweezer
- Very slow to move SLM tweezers
- Fast moving AOD tweezers but inhomogeneous and limited array geometry



AOD's tweezers inhomogeneities
Reproduced from [3]

...for quantum computing

- Need to know precisely the location of each atom to apply gates by addressing individual qubits (2-qubit interactions) [4]
  - Nondeterministic loading poses a serious limitation for large-scale applications [2]
  - Inhomogeneous tweezers increase position uncertainty [4]
- The more qubits we have, the more complex calculations we can perform
- Currently the record is a 324-atom defect-free array with ~37% of success, in ~200 ms. [5, 6]

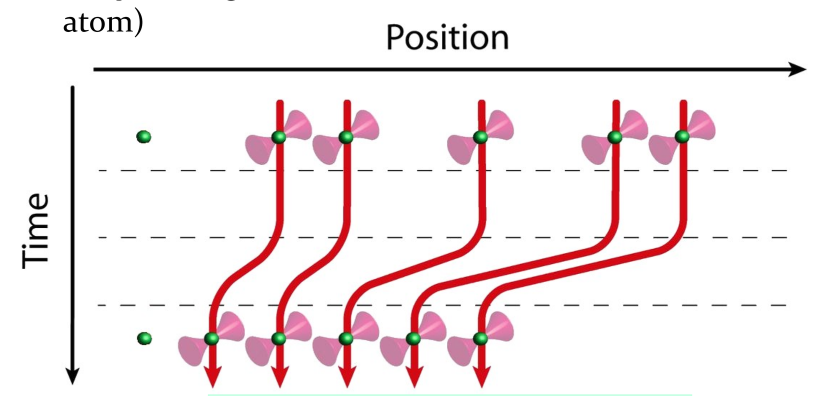


Atom grabbing
Reproduced from [2]

Algorithm

Strategy

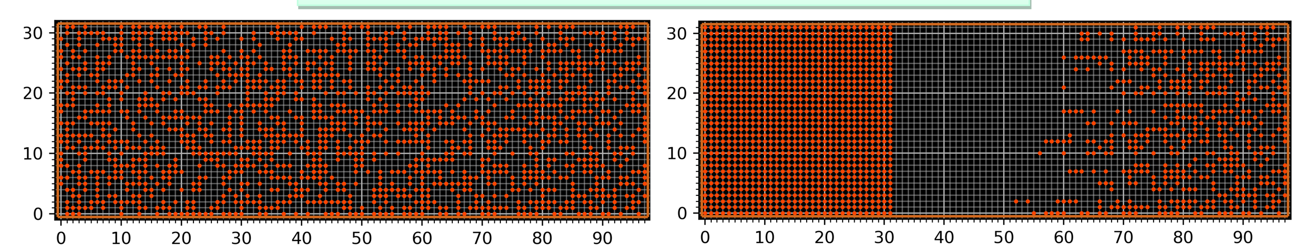
- We start from a rectangle array of tweezers called the *reservoir* and we fill a square subarray
- On all rows we shrink the atoms to the left like in [8]
  - Only moving atoms that need to be moved (reduce the chance to lose an atom)



Shrinking of atoms on a row to the left

- Either we add columns in the *reservoir* to always have enough atoms on each row to fill the square
- Or we make additional moves to fill the remaining holes
- After a second image we can run a fixation algorithm for the atoms lost in the rearrangement

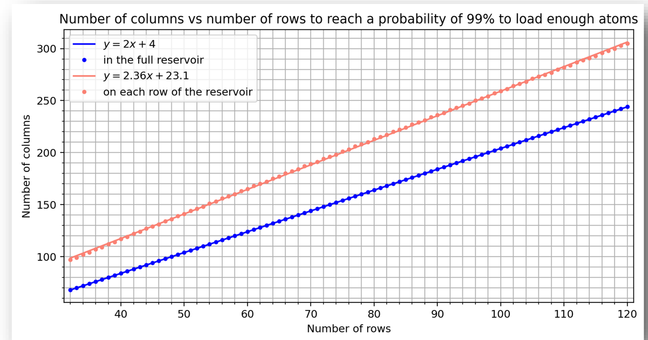
Rearrangement of a defect-free 1024-atom array using a large reservoir

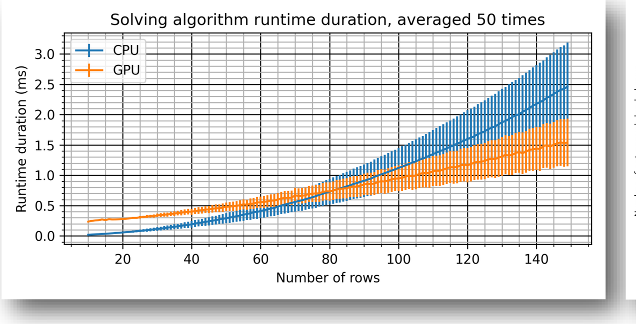


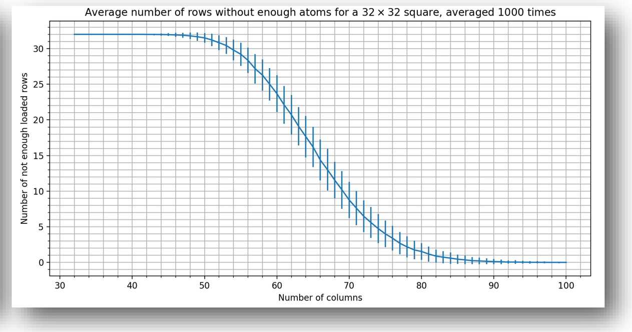
Loading of 50%

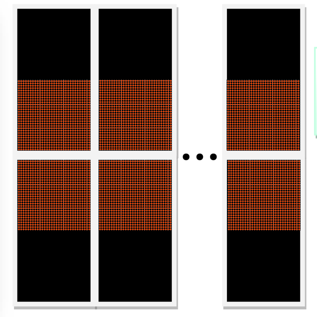
Scalability

- Linear behaviour of probabilities
- Short runtime duration and constant number of moves (one shrink move per row)
- Filling the remaining takes only a few moves
- Ability to combine arrays to create a single large array









Combination of multiple arrays

References

[1] Schymik, Kai-Niklas et al. "In situ equalization of single-atom loading in large-scale optical tweezer arrays". Phys. Rev. A 106. (2022): 022611.

[2] Daniel Barredo, et al. "An atom-by-atom assembler of defect-free arbitrary two-dimensional atomic arrays". Science 354. 6315(2016): 1021-1023.

[3] Robin Eberhard "Preparing next generation tweezer systems for quantum simulation of spin models". Masters thesis, LMU München (2020).

[4] Chew, K.. "Ultrafast energy exchange between two single Rydberg atoms on a nanosecond timescale". Nature Photonics 16. 10(2022): 724-729.

[5] Schymik, Kai-Niklas et al. "Enhanced atom-by-atom assembly of arbitrary tweezer arrays". Phys. Rev. A 102. (2020): 063107.

[6] Schymik, Kai-Niklas et al. "In situ equalization of single-atom loading in large-scale optical tweezer arrays". Phys. Rev. A 106. (2022): 022611.

[7] Bluvstein, Mikhail D.. "A quantum processor based on coherent transport of entangled atom arrays". Nature 604. 7906(2022): 451-456.

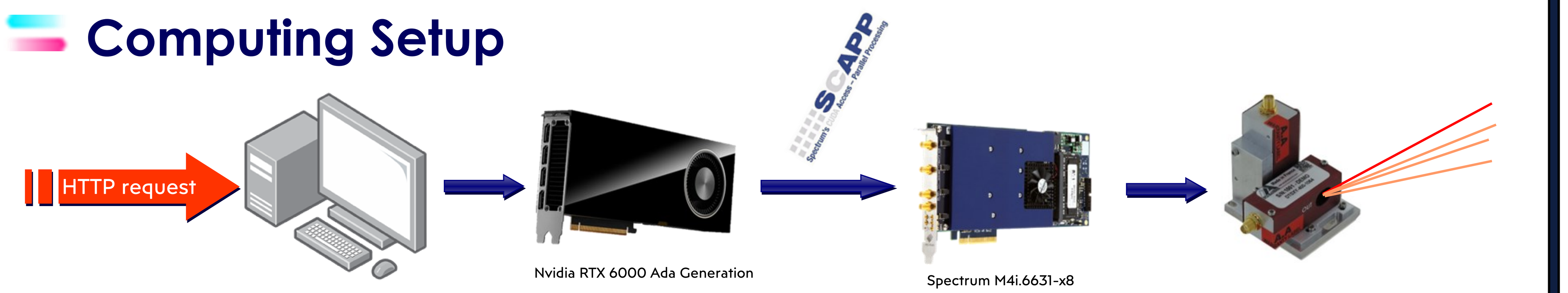
[8] Manuel Endres, et al. "Atom-by-atom assembly of defect-free one-dimensional cold atom arrays". Science 354. 6315(2016): 1024-1027.

[9] Shibasaki, Yukiko et al. "Analysis and Design of Multi-Tone Signal Generation Algorithms for Reducing Crest Factor." 2020 IEEE 29th Asian Test Symposium (ATS). 2020.

[10] Weikun Tian, et al. "Parallel Assembly of Arbitrary Defect-Free Atom Arrays with a Multitweezer Algorithm". Physical Review Applied 19. 3(2023).

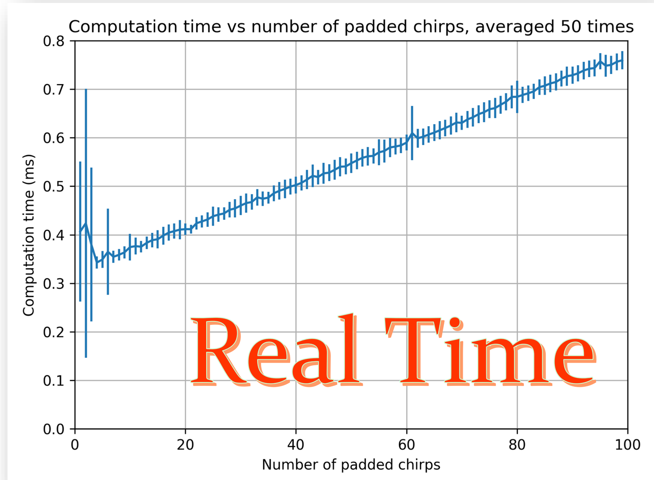
Experimental setup

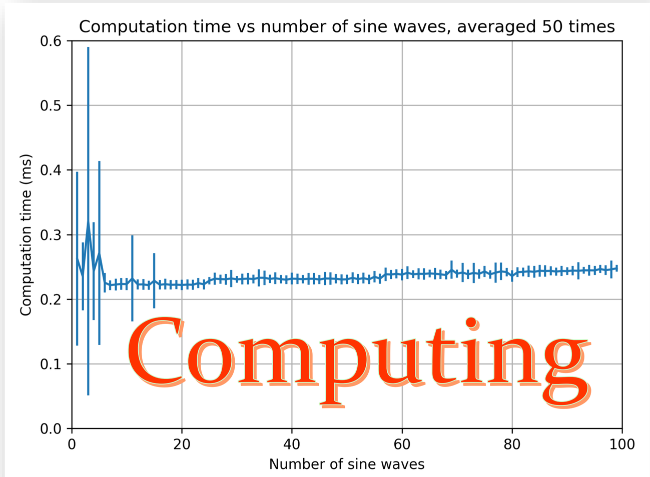
Computing Setup

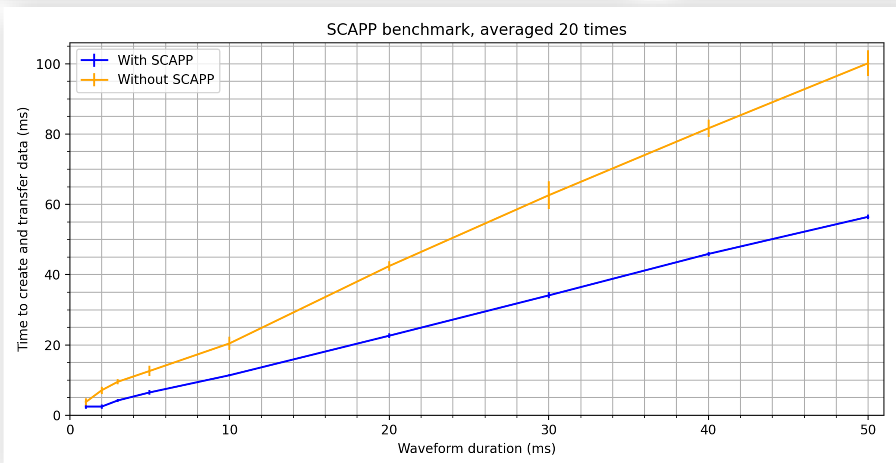


700 MS/s, 700k values, 1 ms waveform. Compute on GPU and transfer back to host memory

- High performance computer
  - AMD Ryzen Threadripper PRO 5975WX 32-Cores
  - 64 GB RAM
  - Nvidia RTX 6000 Ada Generation
  - PyCUDA
- Arbitrary Waveform Generator card
  - Spectrum M4i.6631-x8
  - SCAPP driver relying on Nvidia GPUDirect RDMA for direct transfer
- **No pre-computation**
- Kitayoshi phase to reduce the crest factor [9]
- Constant jerk chirp to move atoms faster [7]
- Variable amplitude for padded chirps

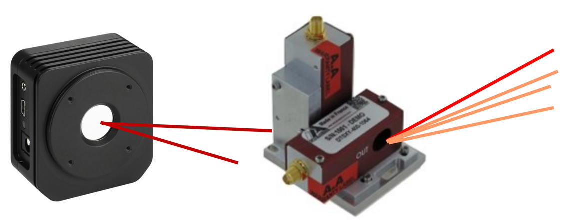




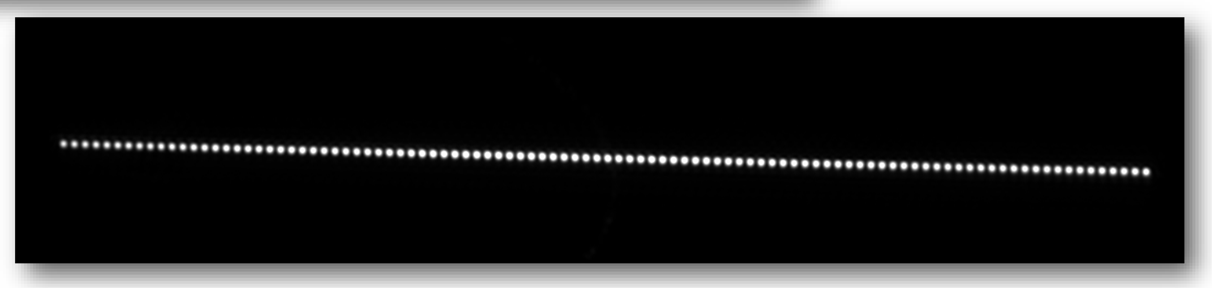


Time to compute and transfer the waveform

Optical Setup



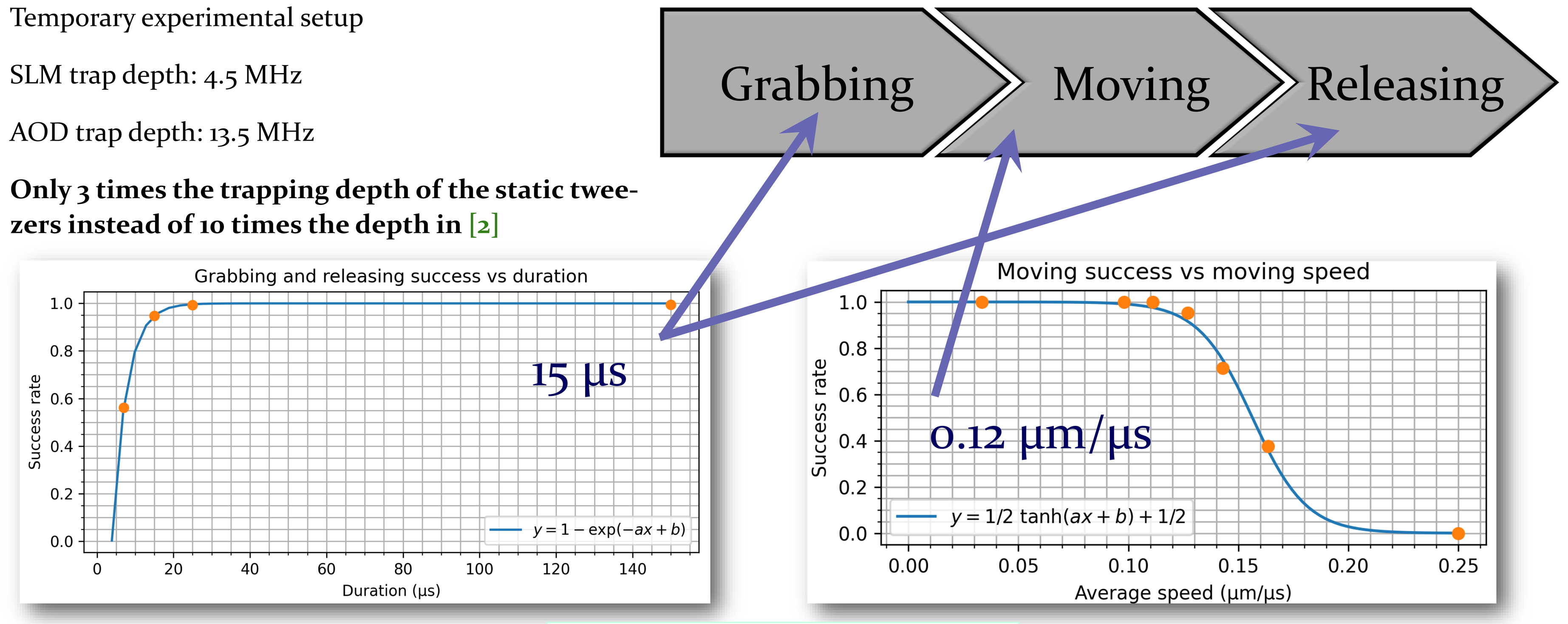
- Thorlabs deformable mirror corrects AOD aberrations
- 850 nm moving tweezers, same as static tweezers but with a perpendicular polarisation



Experimental results

Rearrangement duration

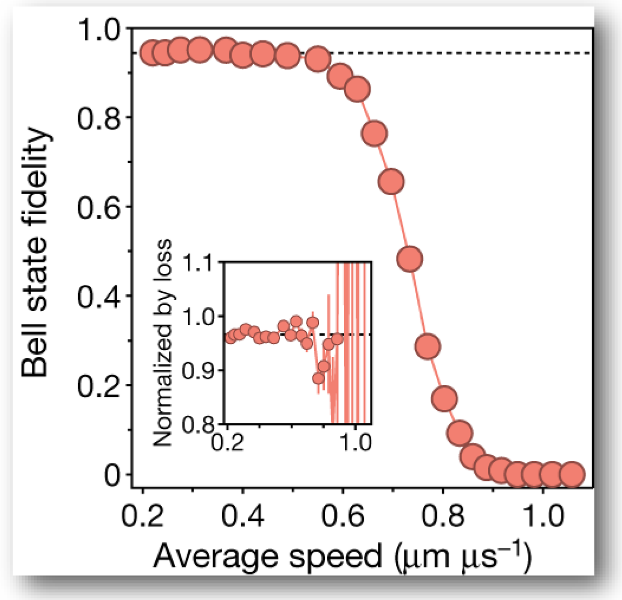
- Temporary experimental setup
- SLM trap depth: 4.5 MHz
- AOD trap depth: 13.5 MHz
- **Only 3 times the trapping depth of the static tweezers instead of 10 times the depth in** [2]



Atoms moved on a distance of 5 µm

- Maximum distance covered by an atom for a 32x32 square to fill with 5-µm gaps and a 98-column reservoir: 330 µm
- 2.75 ms (or half if we centre the square in the reservoir) to move atoms on a row
- 32 rows to shrink
- Total sequence of 89 ms
- We could expect to reduce these times by increasing the trapping depth

Reproduced from [7]



Results of other groups

- Moving speed of 0.55 µm/µs in [7]
- Grabbing speed and releasing speed of 60 µs when up to 20 mobile tweezers are simultaneously turned on in [10]

Early rearrangements

- 86% of success rate for rearrangement in a 3x8 reservoir with moving tweezers fives times deeper than static tweezers

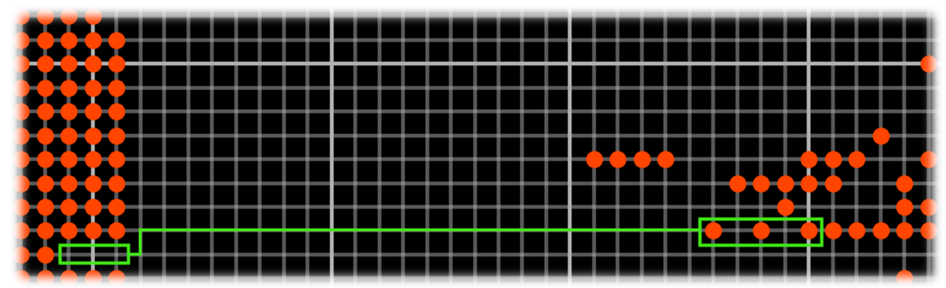
Outlook

Increase array size

- To achieve a defect-free 1024-atom array, we need to use the new powerful laser in the laboratory to be able to create enough tweezers
- To achieve a defect-free 10k-atom array, additional problems would arise due to the size of the array (field of view, angular bandwidth of the AODs and SLM, lifetime of the atoms, ...)

Add moves to increase success rate

- Implement S moves to run the algorithm with fewer additional columns in the reservoir



- Implement L moves for the fixing algorithm in case of atom losses in the rearrangement process (a kind of error correction) combined with a block move to the left