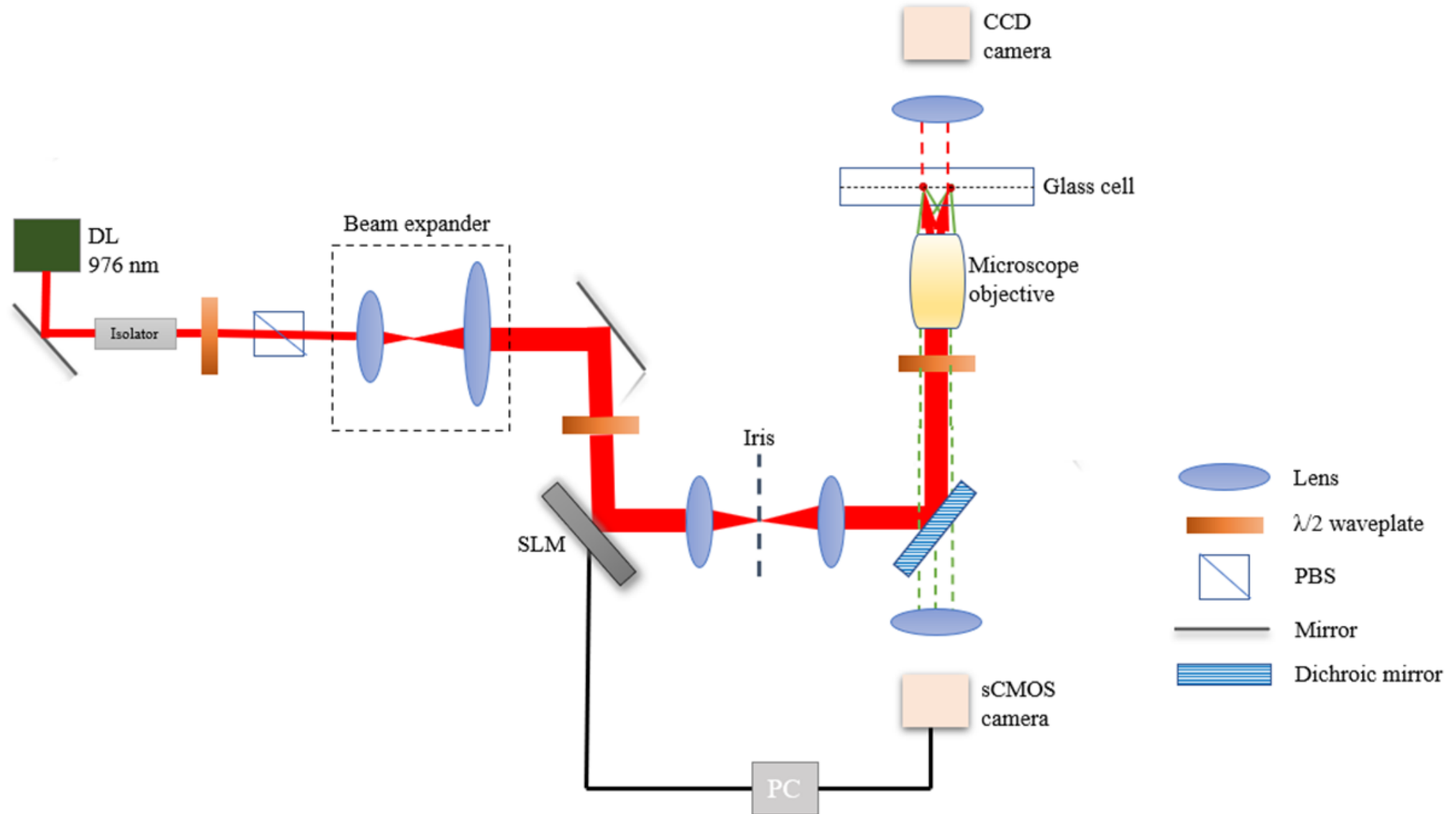
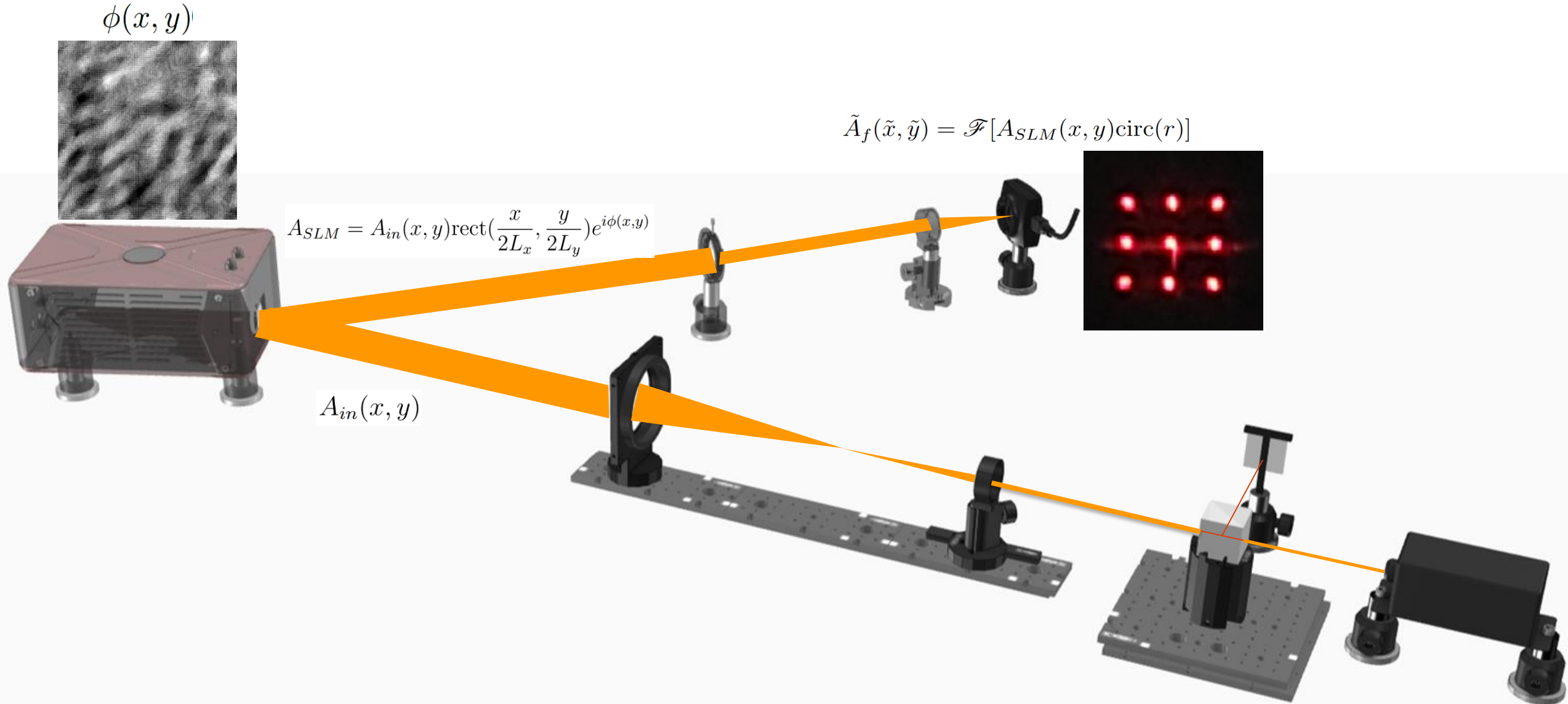
A photograph of a laboratory setup for optical tweezers. The setup is built on a black breadboard with a grid of holes. A red laser beam is visible, reflecting off a mirror and passing through various optical components like lenses and prisms. A black cylindrical component, likely a lens or a filter, is mounted on a white block. A red laser diode is also visible, mounted on a white block. The background is dark, and the overall scene is dimly lit, with the red laser light providing the primary illumination. A semi-transparent blue circle with a pink outline is overlaid on the left side of the image, containing the text.

Design and characterization
of optical tweezers for single
atom trapping

Optical Setup



Generation of Optical Tweezers



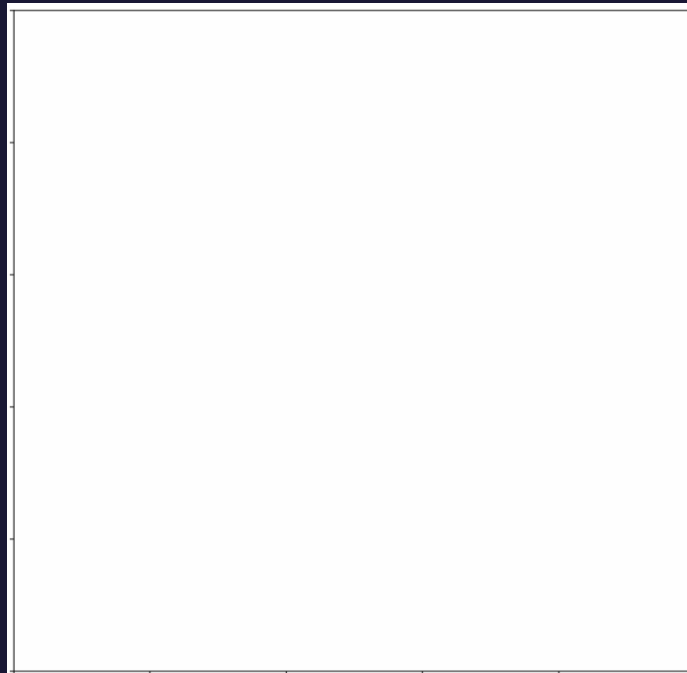
Defect free atom array

Why?

- For quantum computing, we need continuous array of atoms

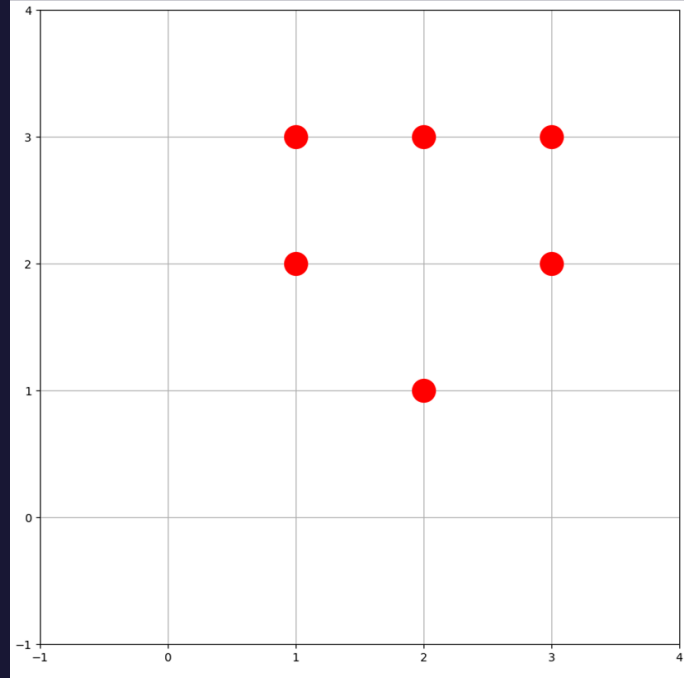
Why we get array with defects?

- Probability of atom getting trapped at each sites is 0.5
- probability for trapping N atoms is $(0.5)^N$, e.g. trapping 9 atoms it is around 0.002



Defect free atom array

- **How to increase probability?**
 - One approach is to design a system to capture atom $>2N$
 - Afterwards, fill vacant sites from nearby reservoir atoms



Defect free atom array

- **Why it has higher probability?**

- Probability is equivalent to $P(N|M)$, where N is #atoms we want out of M traps

$$P(N|M) = \left(\frac{1}{2}\right)^M \sum_{n=N}^M \binom{M}{n}$$

For $N = 9$ and $M = 25$, probability of obtaining defect free atom is about 0.9

Defect free atom array

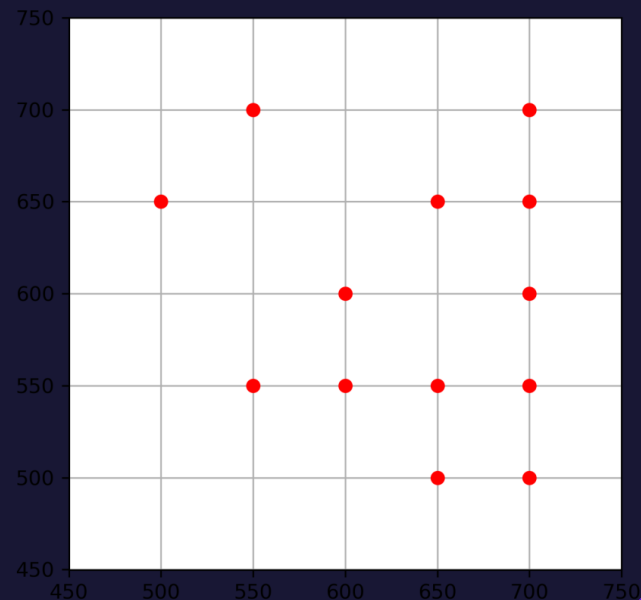
- **How will we approach?**

1. Determine the location of trapped atoms
2. Identify the target positions of atoms
3. Matching between reservoir atoms and voids
4. Path Planning
5. Generation of phase mask and movement of tweezers

Defect free atom array

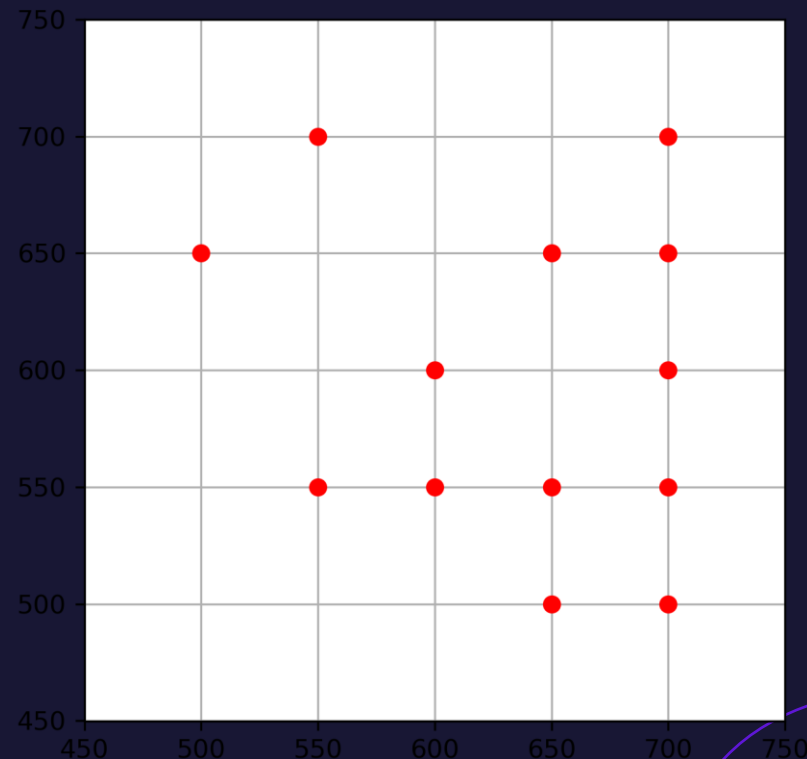
- **Determination of location of trapped atoms**

- In original experiment, image of fluorescence of the atom captured will determine location
- For now, trapped atom configuration is generated randomly
 - Known dimensions n
 - Probability of each site = 0.5
 - Pre defined spacing between sites



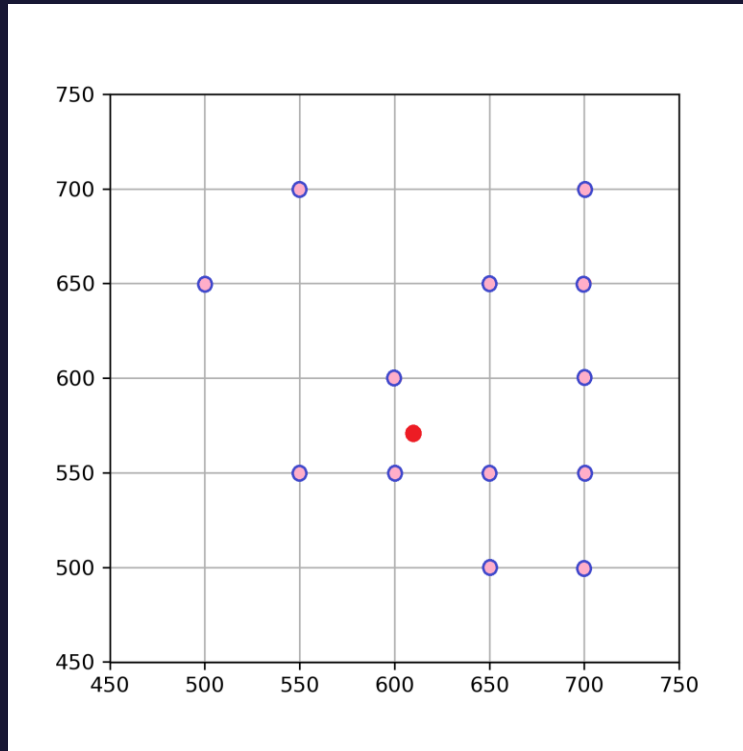
Defect free atom array

- **Identification of target positions of atoms**
 - Optimal position of centre target array position obtained through Centre of Mass (COM) strategy



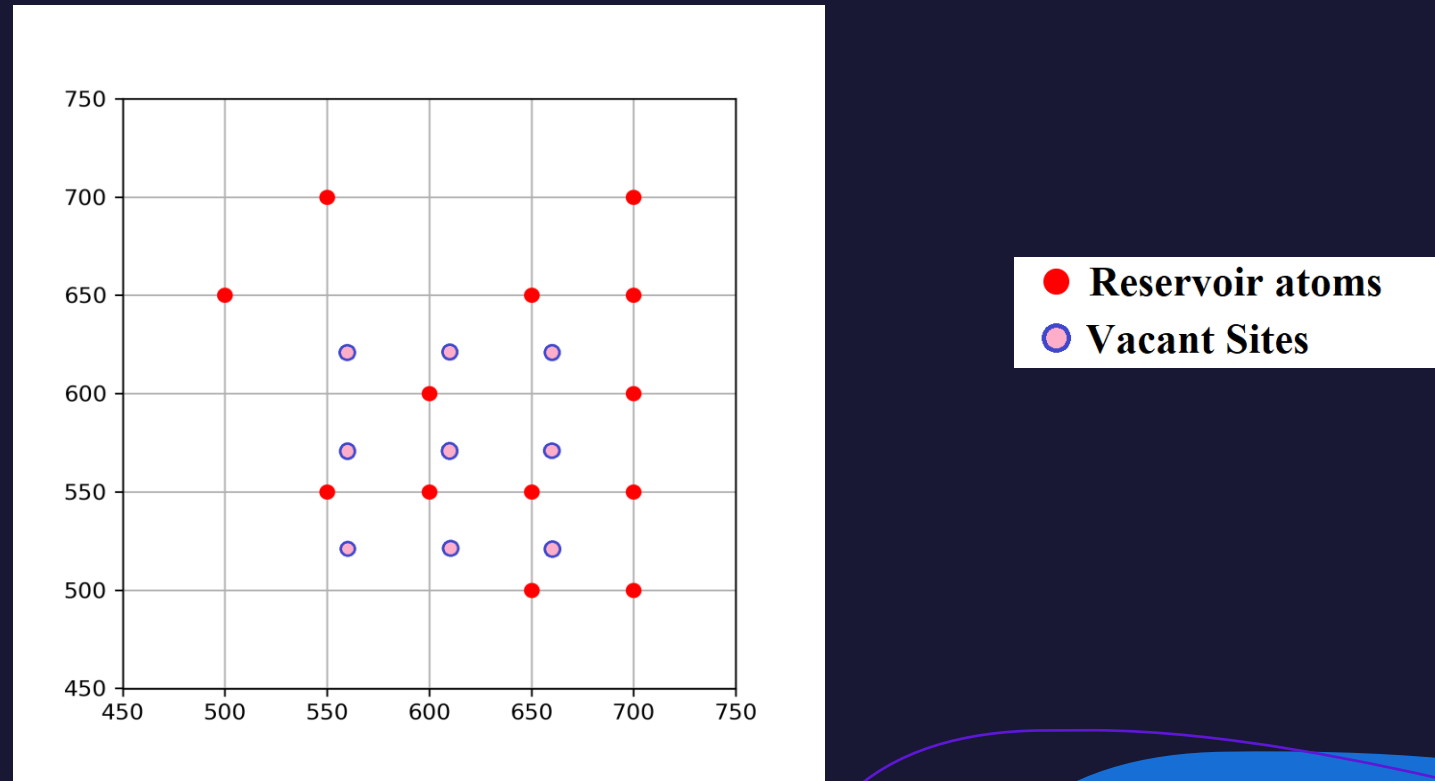
Defect free atom array

- **Identification of target positions of atoms**
 - Optimal position of centre target array position obtained through Centre of Mass (COM) strategy



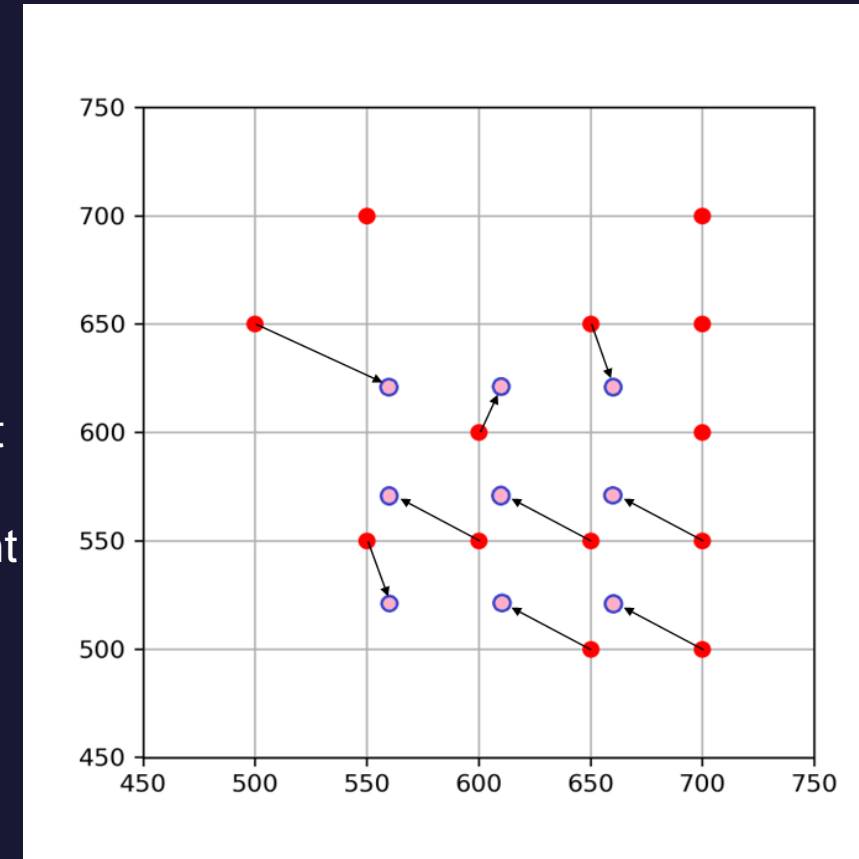
Defect free atom array

- **Identification of target positions of atoms**
 - Optimal position of centre target array position obtained through Centre of Mass (COM) strategy



Defect free atom array

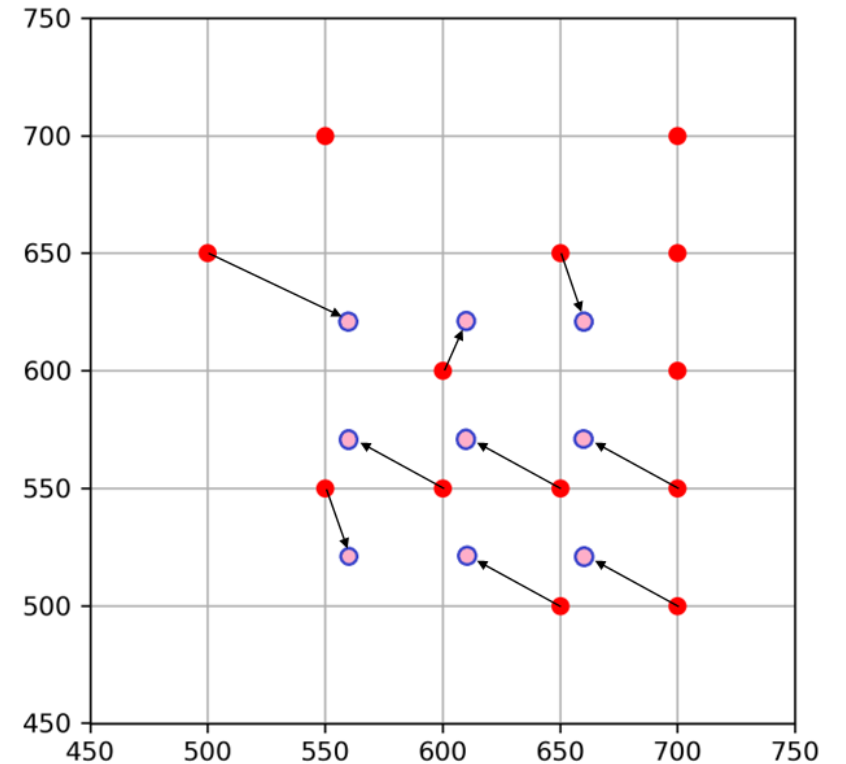
- **Matching between void and reservoir atoms**
 - Matching can be done using
 - Brute Force (Time complexity: exponential)
 - Hopcroft – Karp Algorithm (Time complexity: polynomial)
 - Hungarian Algorithm (Time complexity: polynomial)
 - Hopcroft – Karp Algorithm gives matching without any constraint
 - Hungarian Algorithm gives matching by minimizing the constraint
 - Distance between void sites and reservoir is used as constraint



Defect free atom array

- **Obtaining Path**

- Atom will be moved in straight line
- Speed of movement of tweezers depends on:
 - # frame
 - Frame rate



Defect free atom array

- Movement of tweezers
 - Generation of phase mask for each frame
 - Projecting each phase mask on SLM with desired frame rate

