Property Testing

Property Testing

- Property testing algorithms are algorithms that perform a certain type of approximate decision.
- property testing algorithms are used to decide if some mathematical object (such as a graph or a boolean function) has a "global" property, or is "far" from having this property, using only a small number of "local" queries to the object.
- This task should be performed by inspecting only small part of the whole object.

Property testing

- Need to specify queries that testing algorithm will perform
- Distance measure between objects.

Techniques

- The self-correcting approach
- The enforce and test approach
- Random Walks

Graph Connectivity

- Graph is connected or not ?
- If it is connected then accept it
- If G is “far” from being connected then reject it.

- Graph G is ε far from being connected if at least εm edges must be added to establish connectivity ($m=|E|$)

Algorithm

- If G is ε-far from being connected then G contains lots of small Components.
- Perform a BFS on the randomly selected component
- If BFS terminates then Graph is not connected.

Algorithm

- Repeat 4n/εm times
  - Pick any vertex v Є V UAR
  - Perform BFS from v visiting at most 2n/ εm nodes.
  - If BFS terminates then reject G and Exit
- Accept G

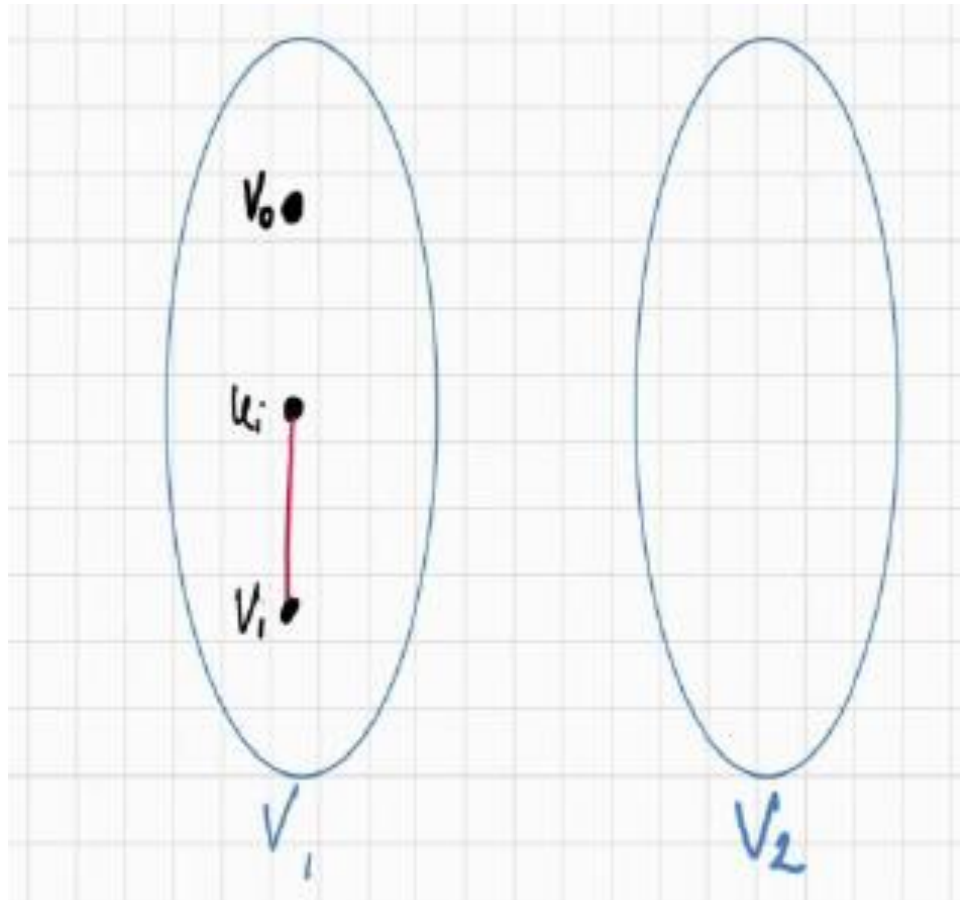
Enforce and Test

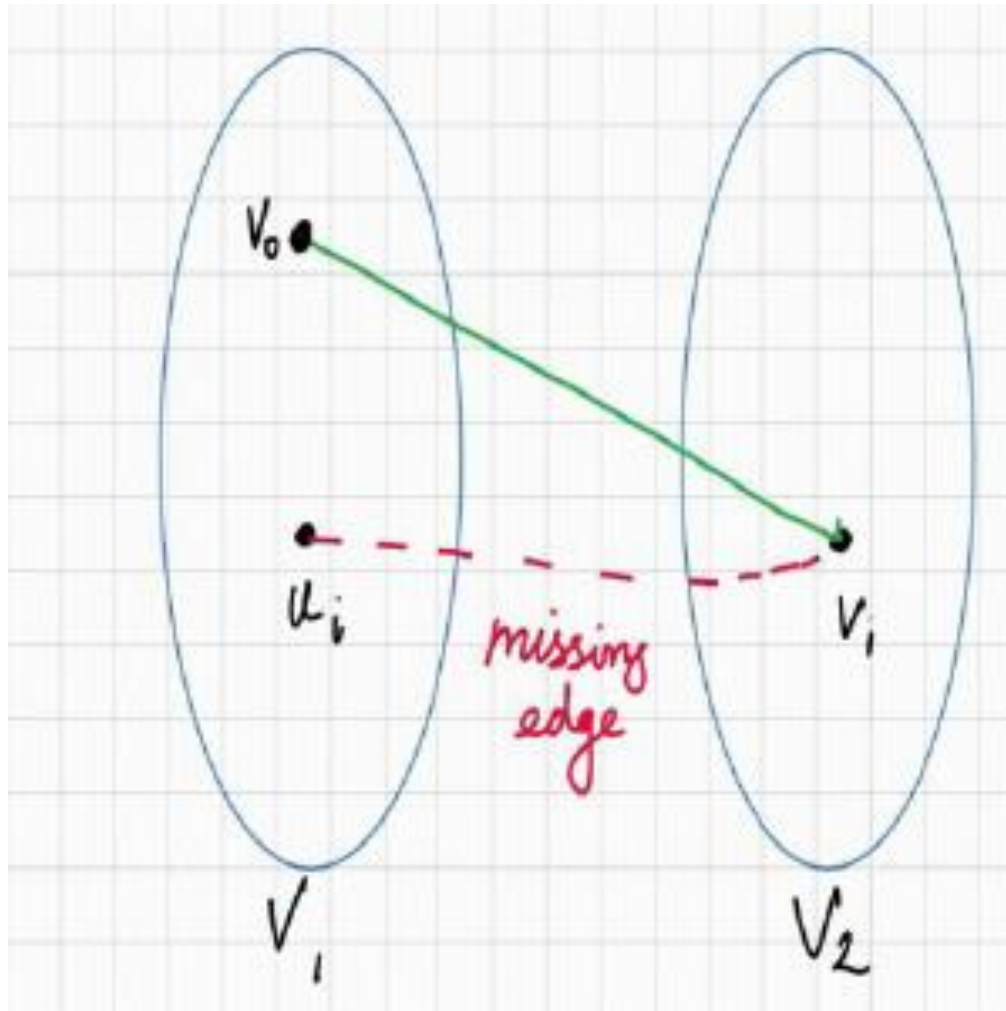
- Testing whether Graph G is biclique?

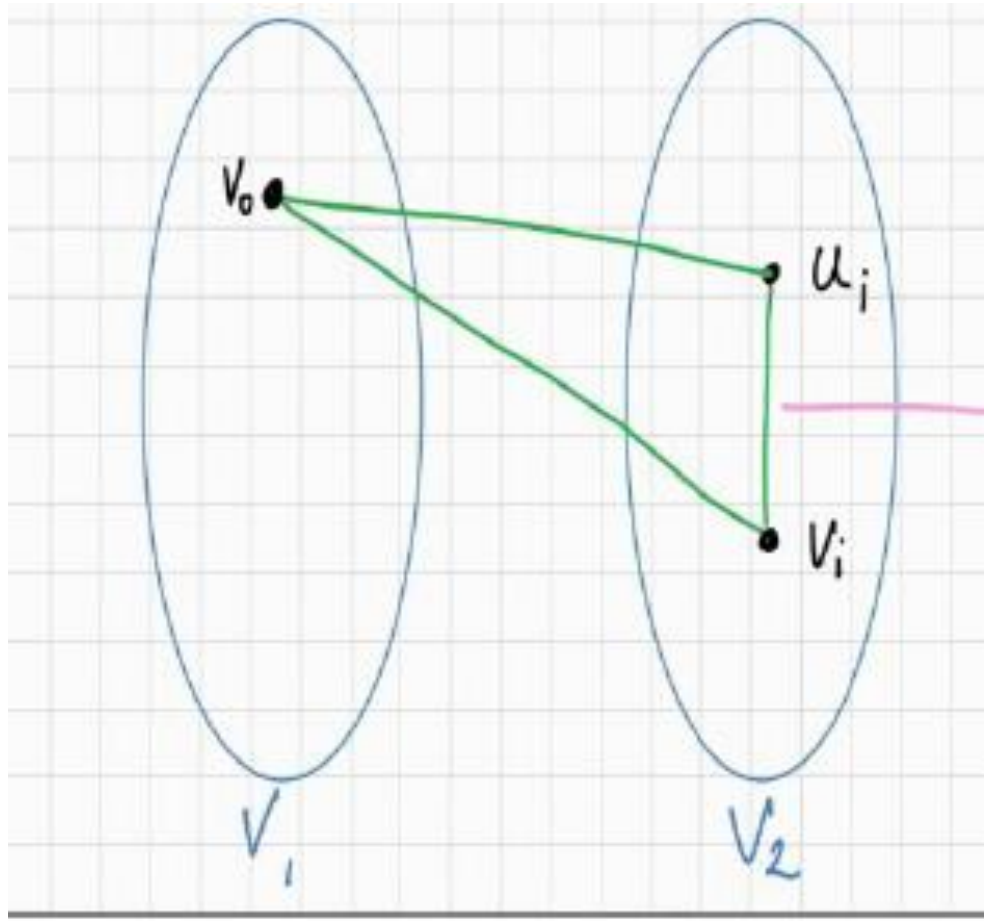
- A Graph G is biclique if there exists a partition (V1,V2) of graph Vertices such that E = V1 X V2

Algorithm

- Enforce : Pick a vertex v_0
- Test :
- for i=1 to 2/ε
  - Pick a pair of vertices u_i , v_i UAR
  - If biclique property is violated then reject
- Accept


V_0
u_i
V_1
V_1
V_2


v_0
u_i
v_i
missing edge
V_1
V_2


v_0
u_i
v_i
V_1
V_2

Random Walk

- Given a graph and a starting point (node), we select a neighbor of it at random, and move to this neighbor;
- Then we select a neighbor of this node and move to it, and so on;
- The (random) sequence of nodes selected this way is a random walk on the graph

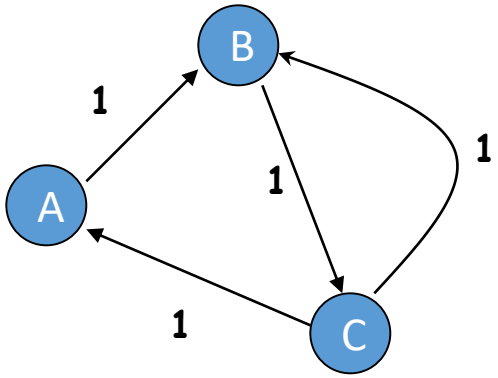
An example

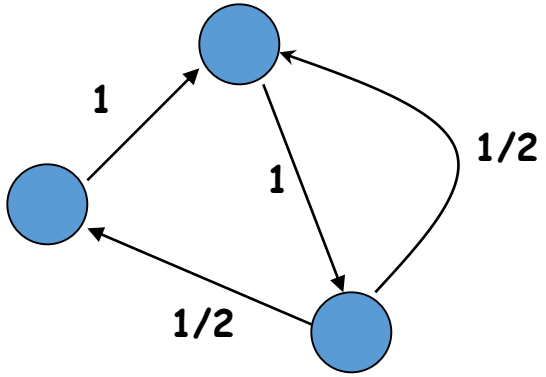
0	1	0
0	0	1
1	1	0

Adjacency matrix A

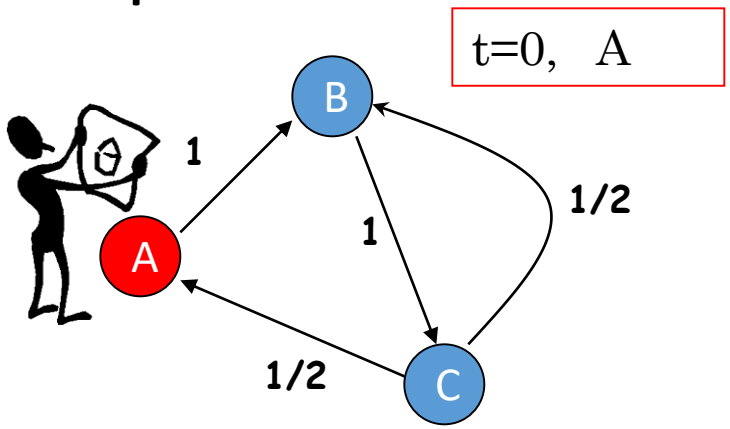
0	1	0
0	0	1
1/2	1/2	0

Transition matrix P

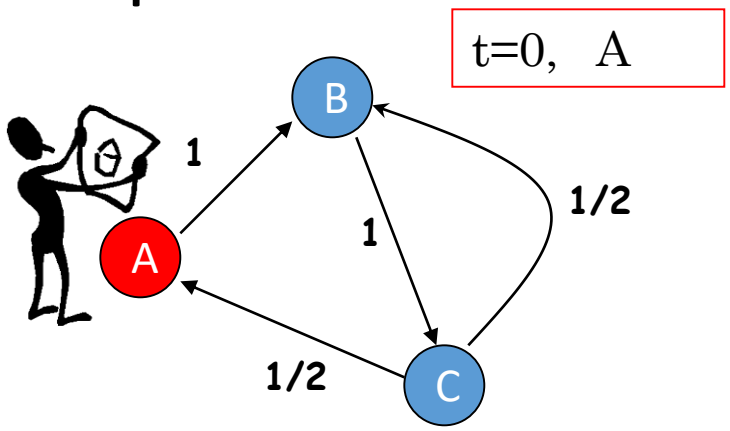


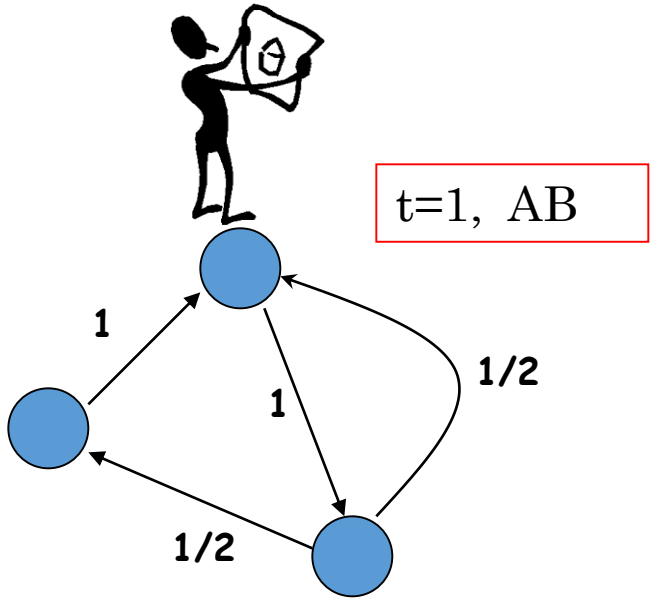


An example

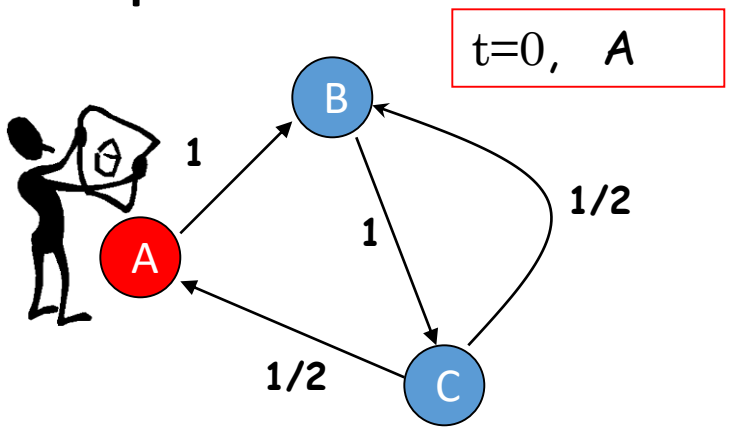


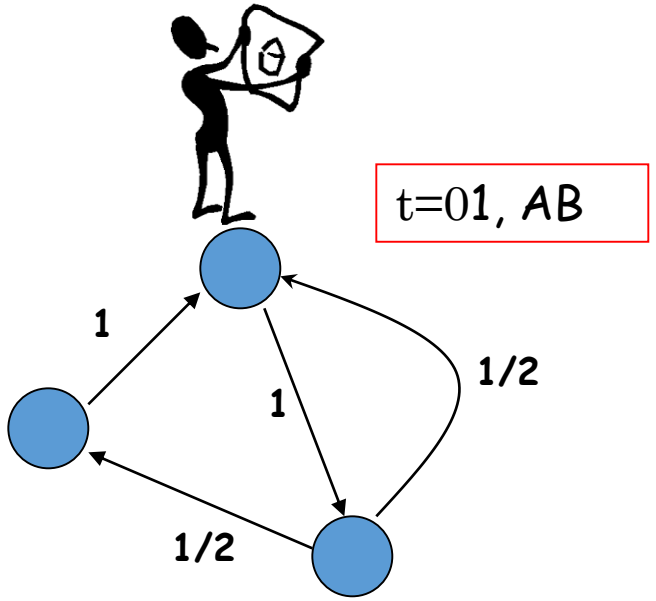
An example

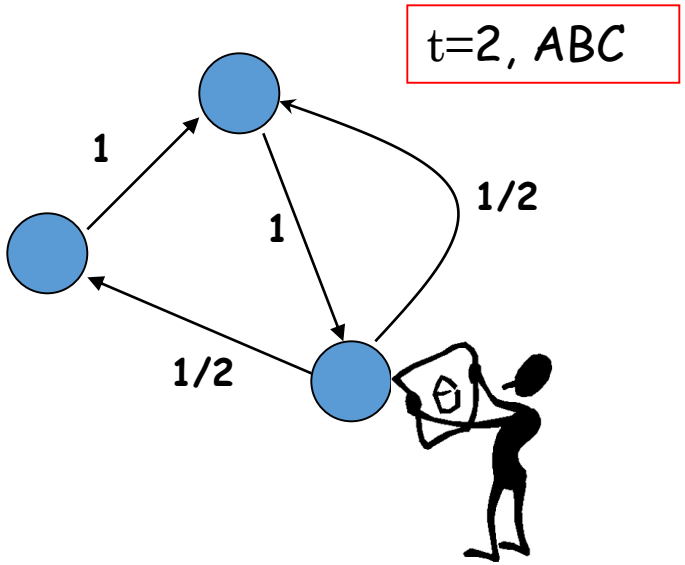




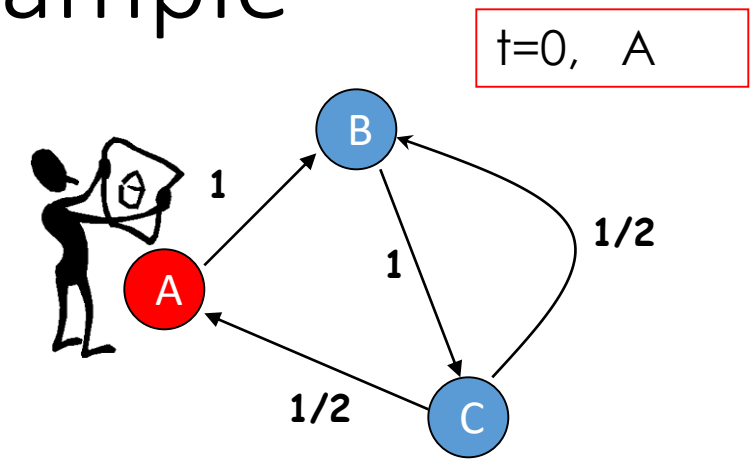
An example

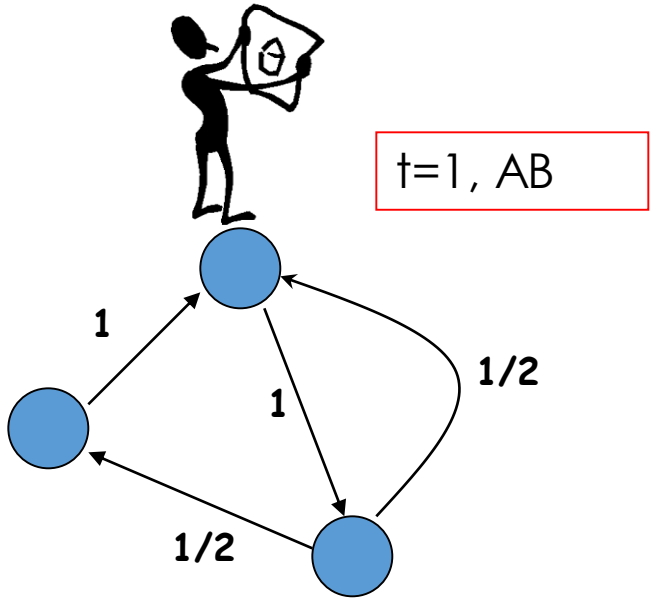


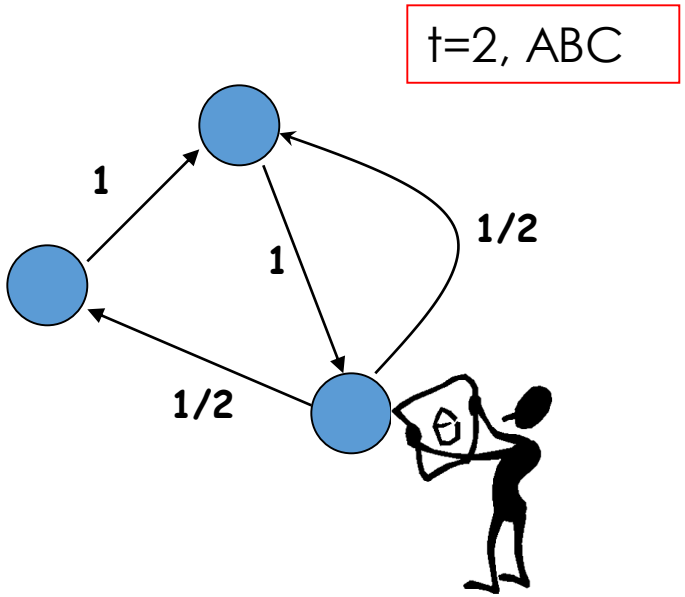


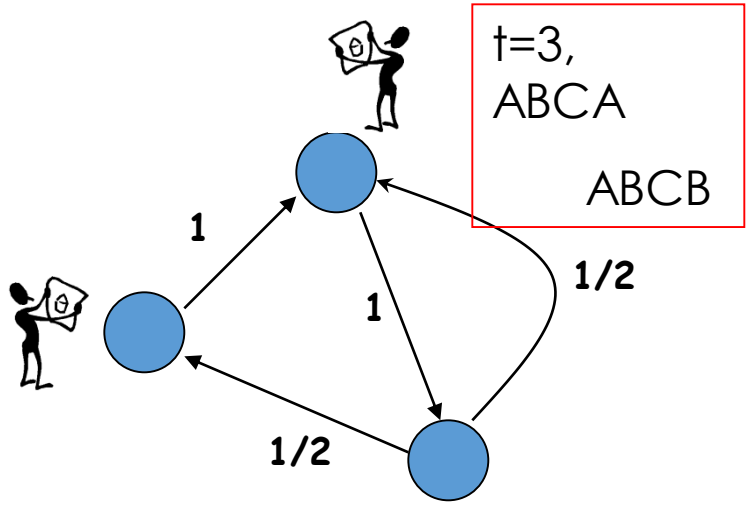


An example









Random Walks on Undirected Graphs

- *A random walk on G is a Markov chain defined by the sequence of moves of a particle between vertices of G. In this process, the place of the particle at a given time step is the state of the system. If the particle is at vertex i and if i has d(i) outgoing edges, then the probability that the particle follows the edge (i, j) and moves to a neighbor j is 1/d(i).*

Why are random walks interesting?

- When the underlying data has a natural graph structure, several physical processes can be conceived as a random walk

Data	Process
WWW	Random surfer
Internet	Routing
P2P	Search
Social network	Information percolation

The Blum-Luby-Rubinfeld Linearity Test

- **BLR Test (Blum, Luby, Rubinfeld).** ***Choose uniformly random points x, y*** $\in \{0, 1\}^n$***. Test if***
- $f(x) + f(y) = f(x + y)$.

- http://nptel.ac.in/courses/106106142/s

Linearity Testing

Linearity Test

1) Uniformly and independently select $\Theta(1/\varepsilon)$ pairs of elements $x,y \in F^n$.
2) For every pair x,y selected, verify that f(x)+f(y) = f(x+y).
3) If for any of the pairs selected linearity is violated (i.e., $f(x)+f(y) \neq f(x+y)$), then REJECT, otherwise ACCEPT.

Query complexity: $\Theta(1/\varepsilon)$, i.e., independent of **n**. In contrast to learning where need $\Omega(n)$ queries/examples.

Theorem: If f is linear then test accepts w.p. 1., and if f is ε-far from linear then with probability at least 2/3 the test rejects it.

Lemma: If f is accepted with probability greater than 1/3, then f is ε-close to linear.

Property Testing

- Is closely related to many areas in theory – communication complexity, Probabilistic Checkable Proofs (PCP), learning theory, coding theory (locally decodable codes), approximation theory, and many more.
- Recently property testing is being used for many real life applications –
  - Google uses it for storing and recovering emails
  - Twitter uses to understand the network of its users,