Pollard's rho factorization method

Random Factoring

- Let $p|n$ and $a \equiv b \pmod{p}$
- Note: p is unknown and a,b is randomly selected
- $a \not\equiv b \pmod{n} \Rightarrow \gcd(a-b,n)$ is a non-trivial factor of n

Proof: $p|(a-b)$, $p|n \Rightarrow p|\gcd(a-b,n)$

- Let $f: S \rightarrow S$ be a random function
- We use f to generate $x_0, x_1, x_2, \ldots$ defined by $x_{i+1} = f(x_i)$.
- Since S is finite, the sequence must eventually cycle.
- Then we can use this sequence to test $\gcd(x_i - x_j, n)$ factors n or not.
- Require $O(\sqrt{n})$ Memory and Time

(birthday problem)

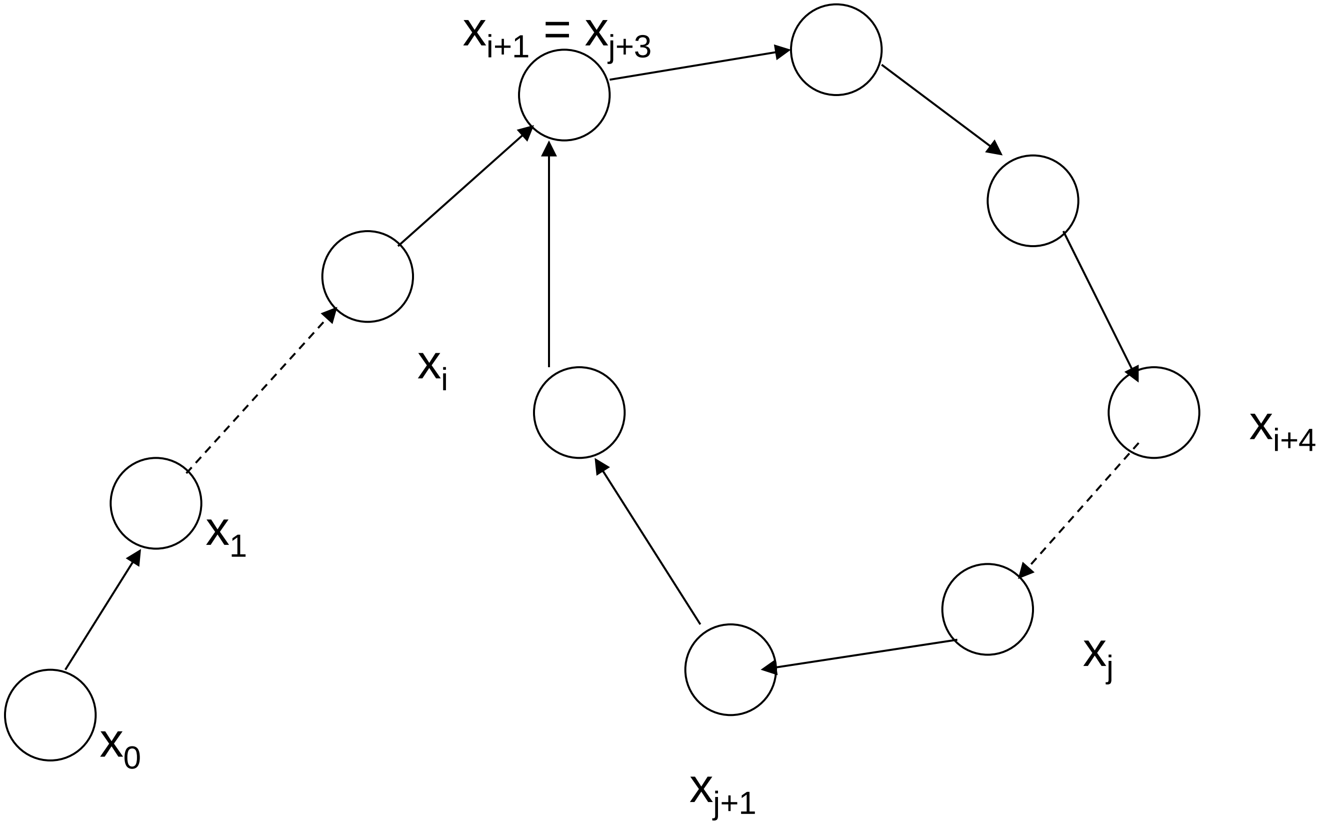
Example

- Let f = $x^2+1 \bmod 15$

x_0	x_1	x_2	x_3	x_4	x_5	x_6	x…
1	2	5	11	2	5	11	…

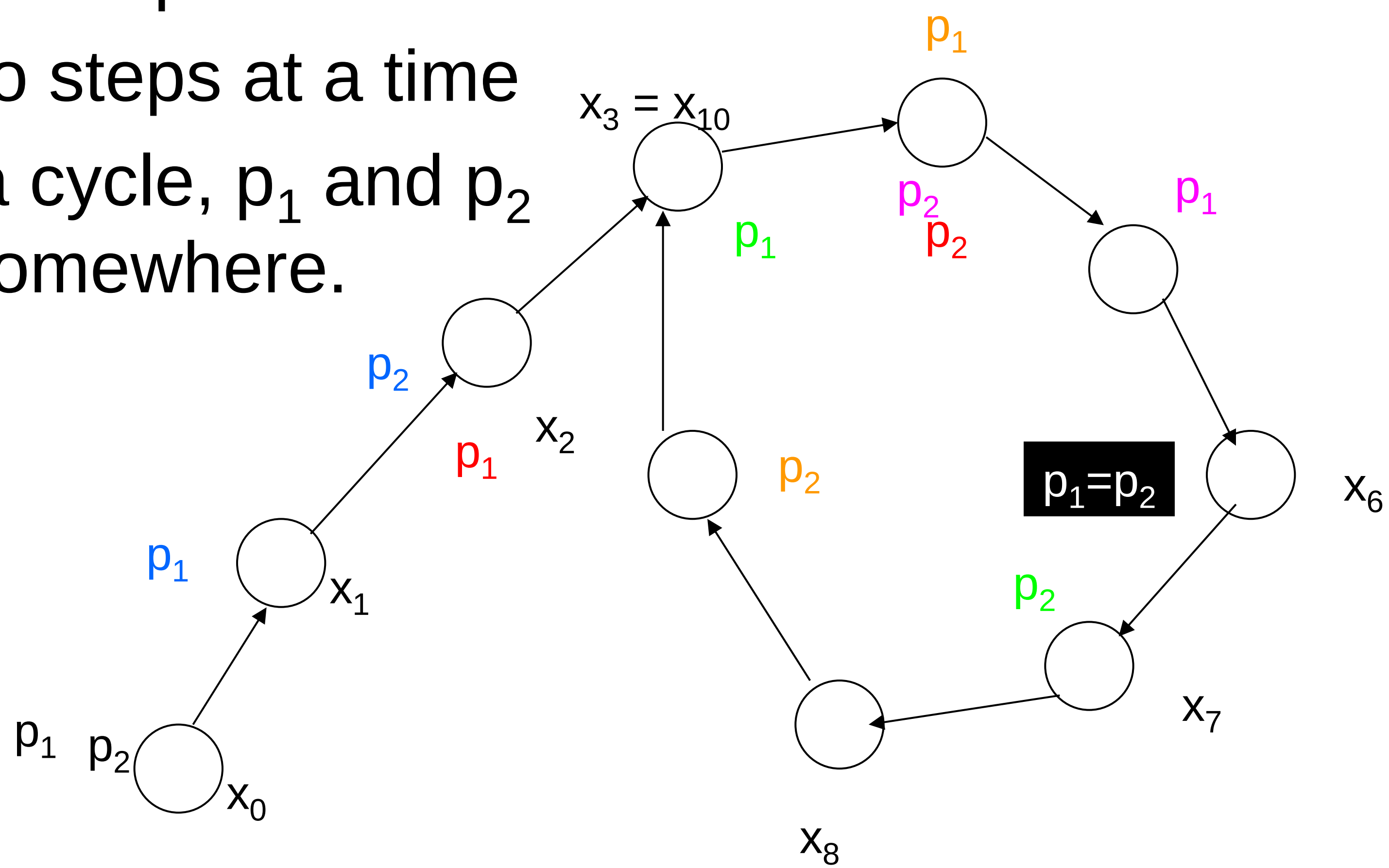
- Memory all x_i and compute every gcd(x_i-x_j,n)
- gcd(5-2,15) = 3

- The sequence of f is cyclic.



Floyd's cycle finding

- Let p_1 and p_2 be two pointer.
- p_1, p_2 starts at x_0.
- p_1 goes one step at a time
- p_2 goes two steps at a time
- If there is a cycle, p_1 and p_2 will meet somewhere.



Pollard’s rho method (1975)

- Combine random factoring and Floyd’s cycle finding (use only tow pointers p_1 and p_2 to save memory).
- Let $f(x) = x^2 + 1 \bmod n$ be the random sequence generator.

Pollard’s rho method

- INPUT: a composite integer n that is not a prime power
- OUTPUT: a non-trivial factor d of n

1. Set $p_1 \leftarrow 2$, $p_2 \leftarrow 2$
2. For i=1, 2, …
   ① $p_1 \leftarrow f(p_1)$, $p_2 \leftarrow f(f(p_2))$
   ② $d \leftarrow \gcd(p_1-p_2,n)$
   ③ If $1<d<n$ then return d
   ④ If $d=n$ then return fail

Example

- n=455459

=613*743

a	b	d
2	2	
5	26	1
26	2871	1
677	179685	1
2871	155260	1
44380	416250	1
179685	43670	1
121634	164403	1
155260	247944	1
44567	68343	743