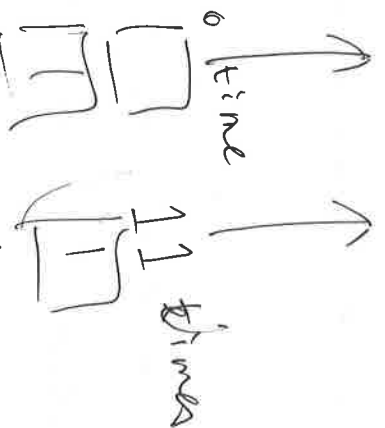


3n nodes

$$(m+1)^2 + (m+1)^2 + (m+1)^2$$



each hyperedge (subset) of size 3 can be represented as a 3n-digit number with 3 1's (in the corresponding positions) and 0's elsewhere.



Take these sequences to represent integers in  $(m+1)$ -ary numbers. then no carry over can happen.

1 idea: Represent each hyperedge/set using an  $n$ -bit string, each bit corresponding to an element. A bit "1" means the element is contained in the set, and "0" if not. Since each set has 3 elements, each such string has 3 ones.

$$\begin{array}{ccccccccc} \boxed{1} & \boxed{0} & \boxed{0} & \boxed{1} & \boxed{0} & \boxed{1} & & & \\ & & & & & = 1 + (m+1)^2 + (m+1)^4 \end{array}$$

$$\begin{array}{ccccccccccc} \boxed{0} & \boxed{0} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{0} & & & & & \\ & & & & & = (m+1) + (m+1)^2 & & & & & \\ & & & & & + (m+1)^5 \end{array}$$


---

Summing up the strings, the corresponding bit in the sum counts the # of times the element is covered by the sets. To cover everything exactly once, we should have an all "1" string.

caveat: We should avoid carries, so we need to work with  $(m+1)$ -ary representation. So the all "1" sequence corresponds

$$\text{to } \sum_{i=0}^{n-1} (m+1)^i = \frac{(m+1)^n - 1}{m}, \text{ a number exponentially large.}$$

$$\text{to } \text{K}$$