Consecutive ones property (COP) in binary matrix

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

With COP

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

Without COP

- A wide range of applications – archaeology (dating artifacts) to biology (DNA sequencing) to computer science
- Interval graph characterization – interval graph's maximal clique-vertex incidence matrix has COP [3]
- Characterizing cubic Hamiltonian graphs – matrix A + I has 2-run COP [4]
- Database consecutive retrieval property testing – COP testing

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Constraint satisfaction problem: Interval Assignment to Set Subsystem[1]

- Set subsystem ~~➔ Intervals: Does there exist a **permutation** of a universal set such that given **subsets permute to intervals**?
- Columns = subsets
- Rows = elements of universe
  - Row indices of consecutive ones form an **interval**
- **COP problem is equivalent to interval assignment** to a set system

Feasible

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

{1, 3}, {1,5}, {1,3,5}, {2,4,5} → {1,2}, {2,3}, {1,2,3}, {3,4,5}

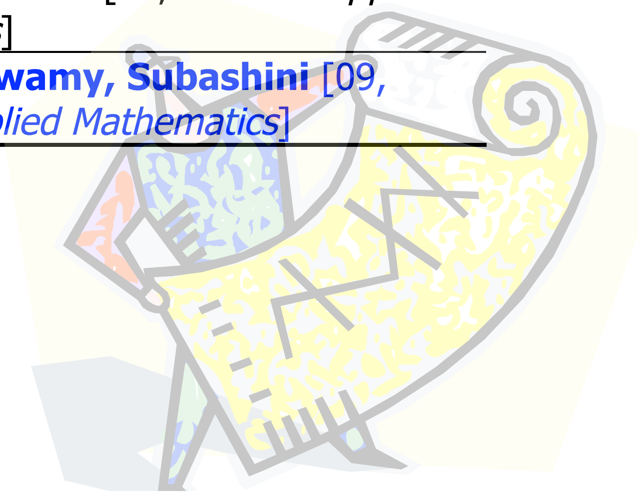
---

[1] A set of subsets of a single universe

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

History

1899	• First mention of COP (archaeology)	**Petrie** [Kendall 69, *Pacific Jounal of Mathematics*]
1951	• Heuristics for COP testing (COT)	**Robinson** [51, *American Antiquity*]
1965	• First poly time algorithm for COT	**Fulkerson, Gross** [65, *Pacific Jounal of Mathematics*]
1972	• Characterization of COP matrices • Forbidden matrix configurations	**Tucker** [72, *Journal of Combinatorial Theory*]
1976	• First linear time algorithm for COT • PQ trees	**Booth, Lueker** [76, *Journal of Computer System Science*]
Circa 2000[2]	• Linear algorithm for COT *without* PQ trees	**Hsu** [02, *Journal of Algorithms*]
2001	• PC trees • A generalization of PQ	**Hsu** [01, *7th Annual International Conference on Computing and Combinatorics*]
1996	• PQR trees • A generalization of PQ to non COP matrices • IDs subcollection of columns preventing COP	**Meidanis, Munuera** [96, *Proc of III South American Workshop on String processing*]
Circa 1998[3]	• Almost linear time algorithm to construct PQR trees	**Meidanis, Telles** [98, *Discrete Applied Mathematics*]
2009	• A New Characterization of Matrices with the Consecutive ones property	**Narayanaswamy, Subashini** [09, *Discrete Applied Mathematics*]



[2] Published 2002
[3] Published 2003

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Conditions (characterization)

Necessity & Sufficiency: *Preserving intersection cardinalities*

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

1. Sort the intervals in increasing order of left end point and break ties using the right end points
   i. Discard identical columns

2. Consider **(P_i,Q_i)**
   i. **P_i** = row indices in i-th column
   ii. **Q_i** = the interval assigned to the i-th column
   iii. *Encodes all permutations in which P_i is mapped to Q_i*

3. Iteratively filter the current set of permutations
   i. Using strictly intersecting pairs
   ii. Pair of intersecting intervals, neither contained in the other

{1}, {3}, {5}, {1, 3}, {2, 4}
{2}, {1}, {3}, {1, 2}, {4, 5}

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Permutations from an ICPIA

$\{1, 3\}, \{1, 5\}, \{1, 3, 5\}, \{2, 4, 5\}$
$\{1, 2\}, \{2, 3\}, \{1, 2, 3\}, \{3, 4, 5\}$

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Proof

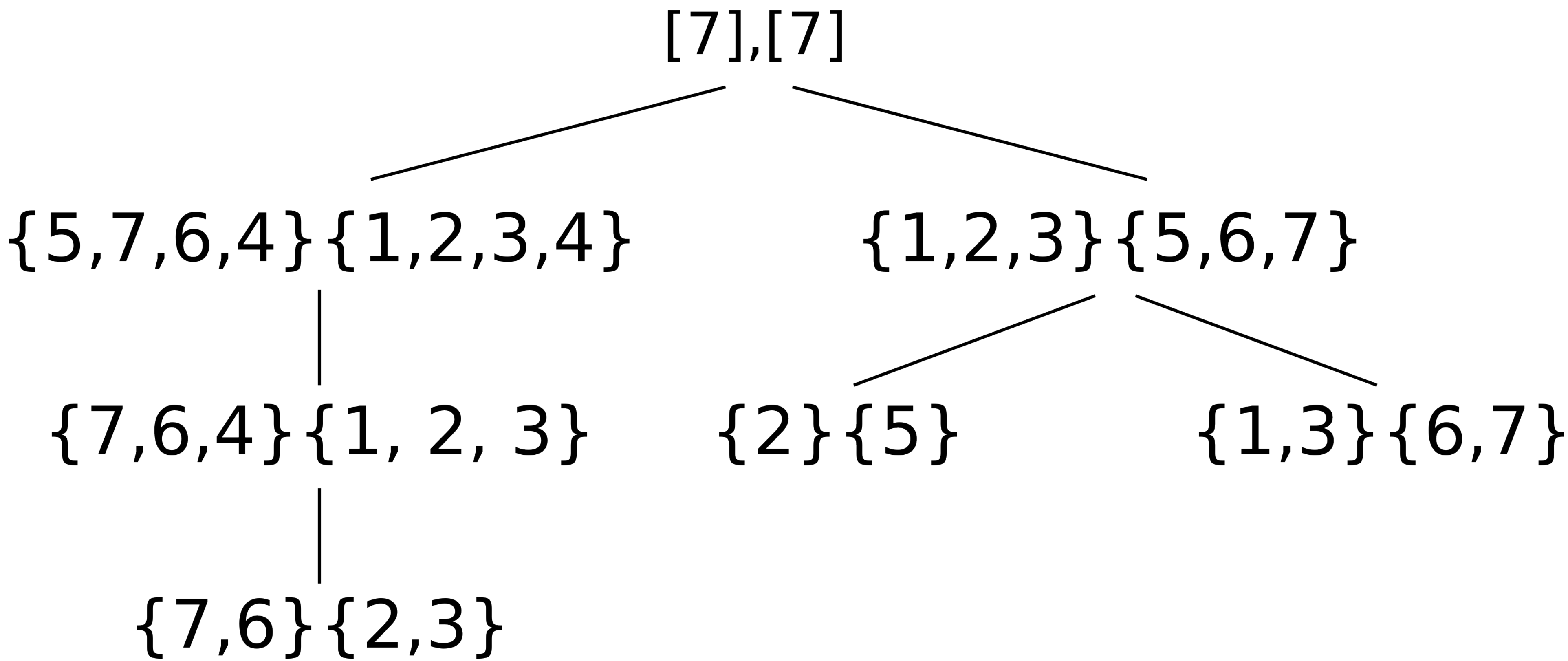
1. Helly property for intervals
    a. For any 3 mutually intersecting intervals one is contained in the union of the other two.
2. Intersection cardinality preserved [1]

Invariants

1. For any (P,Q), (P',Q'), (P'',Q''):
    a. Q is an interval
    b. $|P|=|Q|$
    c. $|P' \cap P''|=|Q' \cap Q''|$
2. At the end no interval is strictly intersecting with another interval
3. Either disjoint or contained

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Getting the permutations - Containment Tree

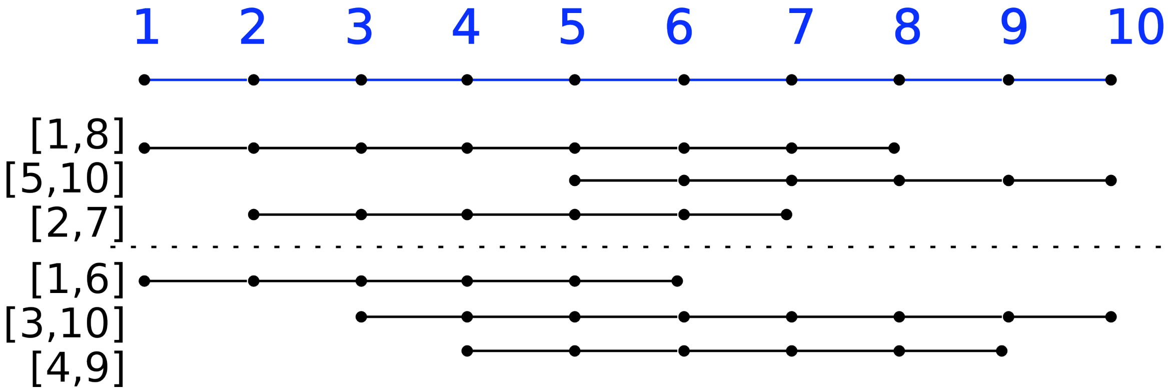


Given an interval assignment, we have a data structure that encodes all permutations, which yield this interval assignment

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Finding a good assignment

For a set of proper intervals and its "**flipped**" the intersection graph are isomorphic
[1,8],[5,10],[2,7] is isomorphic to [1,6],[3,10],[4,9]



To assign intervals to a set system, there are only two choices and these will be decided at the first step

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

1. First Set - left most interval
2. Second set - has strict intersection with first set. So two interval choices
3. Next set (iteratively)-has strict intersection with some interval
4. Exactly one choice of *interval, given intersection cardinality constraints*
5. Failure implies no feasible *interval* assignment
6. Linear time in the number of sets, but computing intersection is costly

Sets left out:

1. Do not have a strict overlap with the sets considered
    a. Disjoint
    b. Contained
2. Two distinct sets are related if they have a strict overlap
3. Consider connected components in this undirected graph

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

1. Each component is a sub-matrix formed by the columns
2. Two components are either
    a. Disjoint
    b. Or all the sets in one are contained in a single set of the other.
3. *An interval assignment to each component implies an interval assignment to the whole set system*

4. Given that an interval assignment to each of the components is feasible.
5. Containment tree/forest on the components
    a. An arc between vertices corresponding to two components if the sets of one are all contained in one set of the other
6. Construct the interval assignment in a BFS fashion starting from the root of each tree

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)

Extensions and questions posed

1. Natural extension: **paths to trees** (intervals correlate to paths)
    a. Characterization of tree path assignment – is there some nice property here?
        i. Is an extra condition of 3-way intersection cardinality preservation sufficient?
    b. What can be its applications?

2. **K-run** problem – instead of all ones consecutive, are there at most k blocks of consecutive ones?

anju srinivasan zabil, iit madras (**anjuzabil@gmail.com**)