

Compatibility Relation and its Application to Network Segmentation and Decentralization

D. Singh (Former Professor, IIT Bombay)
 Department of Mathematics, Ahmadu Bello University,
 Zaria, Nigeria
 mathdss@yahoo.com

T.O William-west (research scholar)
 westtamuno@gmail.com

Abstract - In this paper, some new properties of compatibility classes of a finite set endowed with a suitable compatibility relation are described. An algorithm to compute maximal compatibility classes is constructed and an application of maximal compatibles to network segmentation and decentralization is demonstrated.

Keywords: Compatibility relation, compatibility classes, maximal compatibility classes, minimal covering, network segmentation, network decentralization

I. INTRODUCTION

A relation on a set which is reflexive and symmetric is called a *compatibility relation* (CR). It is difficult to trace since when the profound mathematics of *compatibility relation* came into existence. To our knowledge, Kurepa [12] seems to be the earliest full-blown mathematical exposition on the study of reflexive symmetric relations and graphs. Since then, a number of works ([16], [23], [15], [21], [5], [6], [7], [8], [11], [17], [4], [20], [14] and [19]; etc.), dealing with fundamentals of compatibility relation as well as its applications, have appeared.

However, by now, it can be said that the actual import of compatibility relation has gone far beyond its ordinary linguistic connotation and even that of its mathematical characterization. It has found several applications in different fields of knowledge. Essentially, from application points of view, compatibility relation is useful in solving a class of minimization problems, particularly when the problems are incompletely specified:

- (a) In switching theory, particularly for incompletely specified problems [16, 19];
- (b) In incompletely specified sequential machines (ISSM) for reduction of the number of internal states [6];
- (c) In designing of a class of digital control Units [2, 7];
- (d) In graph theory [8];
- (e) In solving some combinatorial problems, such as scheduling of traffic control [6];
- (f) In phonology [17];

just to mention a few.

In this paper, we propose an alternative algorithm for computing MCCs and its application to network segmentation and decentralization.

II. COMPATIBILITY AND ITS CHARACTERISTIC PROPERTIES

Let S be a set with n elements, usually annotated n -set. A family $\{A_1, A_2, \dots, A_n\}$ of non-empty subsets of S is called a *covering* of S if $S = \bigcup_{i=1}^n A_i$. A cover of a finite set S is called *minimal* if none of its proper subclasses covers S . Note that A_i 's are not necessarily disjoint and hence, it may not define a partition.

Notationally, a *compatibility relation* is sometimes denoted \approx . Also, if R be a compatibility relation on a set S , then $x, y \in S$ are called *R-compatible* or simply compatible to each other if xRy i.e., x is an R -relative of y . Note that compatibility relation, not being necessarily transitive, may not define a partition. However, it does define a covering ([18], for details). Essentially, a compatibility relation defined on a finite set decomposes the set into its possibly pairwise non-disjoint subclasses, henceforth called *compatibility classes* (CCs). It follows that the elements of a CC are pairwise compatible (PC). Note also that every subclass of S may not be a CC, that is, elements of such subclasses are non-pairwise compatible (NPC).

Let S be an n -set $\{x_1, x_2, \dots, x_n\}$ and R a compatibility relation on S . The x_i 's may be representing *nodes* in a network system or *states* in a finite machine or *vertices* in a graph. A subclass $M \subseteq S$ is called a *maximal compatibility class* (MCC) if any element of M is compatible to its every other element and no other element of $S - M$ is compatible to all the elements of M . Equivalently, a compatibility class of S is maximal if it is not a proper subclass of any other compatibility class of S . Graphically, MCCs for a given compatibility relation R can also be viewed as the largest *complete polygons* in the graph of R . A polygon in which every node is connected to its every other node is called a *complete polygon*. A triangle is always a complete polygon and, for a quadrilateral to be a complete polygon, we need both the diagonals. Note that a complete polygon is a CC, which need not be an MCC, unless it satisfies the criteria of

being so. Also, any element of the set that relates only to itself is an MCC, and any two elements of S which are compatible to one another but to no other elements of S form an MCC.

The following results hold:

- (i) There must exist a family $\{A_1, A_2, \dots, A_n\}$ of non-empty CCs of S such that $S = \bigcup_i A_i$.
- (ii) There must exist a family $\{M_1, M_2, \dots, M_k\}$ of MCCs of S such that $\bigcup_{i=1}^k M_i = S$. Moreover, the least number of M_i 's that cover S constitute a minimal cover of S . It is easy to see that the number of minimal covers may be more than one.
- (iii) Every element of S must be an element of at least one of the MCCs of S . Also, whenever $x_i R x_j$, then $\{x_i, x_j\}$ must be a subclass of at least one MCC of S . Note, however that, in general, for a pair of compatible nodes, the inclusion of one of the nodes in an MCC does not necessarily imply the inclusion of the other.
- (iv) Only CCs can be MCCs. More explicitly, for any compatibility class C of S , either C itself is an MCC or C is a subclass of some other MCC of S . It follows that a CC of S with a maximum cardinality is an MCC. Note that there may be more than one CC of the same cardinality and none or some or all of them are MCCs
- (v) Let $\{M_1, M_2, \dots, M_k\}$ be a family of all MCCs of S , then M_i 's are pairwise incomparable with respect to \subseteq (subsethood).
- (vi) S has at least two MCCs if and only if, for $x_i, x_j \in S$ such that $x_i R x_j$, $x_i \in M$ but $x_j \notin M$, where M is an MCC of S .
- (vii) $1 \leq \#\{\text{MCCs}\} \leq n$, where $\#$ denotes cardinality of a set.
 Proof. The number of MCCs is exactly 1 or n according as all the nodes are connected to each other or no two nodes are connected.
- (viii) $\#\{\text{MCCs}\}$ is directly proportional to $\#\{\text{CCs}\}$, and inversely proportional to $\#\{\text{connections amongst the states of an ISSM}\}$.
 Proof follows from definitions given above
- (ix) $\#\{\text{MCCs}\} \leq \#\{\text{CCs}\} \leq \#S = \#\{\text{CMCCs}\}$.
 Proof follows from definitions given above.
- (x) If there exist MCCs containing 1 or 2 elements alone, then they must be part(s) of every (minimal) cover.

Let us consider an example to bring some of the aforesaid observations home.

Let S be $\{x_1, x_2, \dots, x_5\}$, and R be a compatibility relation on S whose simplified graph of is represented in figure 1 below.

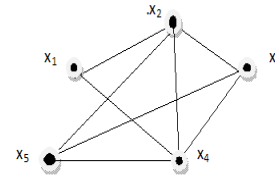


Figure 1: Simplified graph of R (Since R is reflexive and symmetric, loops at nodes are not drawn and only one of $x_i R x_j$ and $x_j R x_i$ is drawn).

It may be observed that $x_2 R x_3 \wedge x_2 R x_4 \not\Rightarrow x_2 R x_5$. From the diagram, it follows that

$C_1 = \{x_1, x_2, x_4\}$, $C_2 = \{x_2, x_3, x_4\}$, $C_3 = \{x_2, x_4, x_5\}$, $C_4 = \{x_3, x_4, x_5\}$ and $C_5 = \{x_2, x_3, x_4, x_5\}$ are the CCs or complete polygons of S . The class $\{x_1, x_2\}$, for example, is not a CC of S . Clearly, $M_1 = \{x_1, x_2, x_4\}$ and $M_2 = \{x_2, x_3, x_4, x_5\}$ are the only MCCs or the largest complete polygons, and these classes define a unique minimal covering of S . Observe that $x_2 R x_3$ holds, and x_2 belongs to both M_1 and M_2 , but x_3 belongs to M_2 and does not belong to M_1 .

Theorem: Corresponding to every minimal cover, there exists a unique compatibility relation.

Proof. Let R be a compatibility relation on an n -set S , and $\Gamma = \{M_1, M_2, \dots, M_k\}$ be a minimal cover of S . Now, since each element of $M_i \in \Gamma$ is R -related to every other element of M_i , and not R -related to all the elements of $M_j, i \neq j$, it follows that all the elements of $M_i \times M_i$ are elements of $R, i = 1, 2, \dots, k$. Then $R = \bigcup_{i=1}^k M_i \times M_i$.

It is immediate to see that R is a compatibility relation. Let us consider the example of section II. Now, $\Gamma = \{M_1, M_2\}$ where $M_1 = \{x_1, x_2, x_4\}$, $M_2 = \{x_2, x_3, x_4, x_5\}$. $R = (M_1 \times M_1) \cup (M_2 \times M_2)$.

$$R = \{\{x_1, x_2\}, \{x_1, x_4\}, \{x_2, x_3\}, \{x_2, x_4\}, \{x_2, x_5\}, \{x_3, x_4\}, \{x_3, x_5\}, \{x_4, x_5\}\},$$

after leaving out loops at each node and admitting only one of symmetric pairs. Moreover, an MCC with a single element is to be included in the end.

It is easy to see that the simplified graph of R is the figure 1 of section II. The converse of this results is not true can be seen from (ii). Note that this theorem has a precedent in partition calculus.

III. THE COMPATIBILITY MATRIX (CM) OF A COMPATIBILITY RELATION R AND CONSTRUCTION OF AN ALGORITHM TO COMPUTE CCs/ MCCs

Abstracting from [16], [5] and [1], we present a simplified algorithm to compute CCs/ MCCs as follows.

In order to construct the CM of R , for each pair of nodes (x_i, x_j) , a one (1) is assigned to it if $x_i R x_j$ and, a zero if it is not the case that $x_i R x_j$. Consequently, the table for the CM has an all – 1 leading diagonal (due to reflexivity of R). Moreover, since the CM of R is symmetric, it is sufficient to tabulate only the elements of its lower triangular parts. However, for a better comprehension, we will consider the complete matrix for discussion in this paper.

Step1: Compute all CMCCs using matrix table (as in [1], but without columns for their *generators* and *weights*).

Step2: Begin with the CMCC of row 1. If it is of cardinality 1 or 2, list it in the column of CCs. The same holds for all other rows of the compatibility table.

Step3: If the CMCC in consideration is of cardinality greater than 2, check whether or not there exists a zero (0) at any intersection of rows and columns of the submatrix generated by the constituent states of that CMCC i.e., the submatrix is an all-1 submatrix or not. If the submatrix is all-1, list it in the column for CCs.

Step4: If the matrix in step 3 is not all-1, ignore the row(s) and column(s) containing the largest number of zeros and list the states which produce the truncated matrix, which is an all-1, in the column for CCs. If not, repeat step 4 until an all-1 matrix is obtained. In such cases, there may be two different CCs.

Step5: Similar to all other existing algorithms, proceed as above for all the rows exhaustively.

Step6: Once the column of CCs are completed, delete all CCs which are proper subsets of some other CC. Also, if repeated CCs occur, delete all such CCs except one of them. The remaining CCs are distinct MCCs.

This approach takes care of deleting superfluous states of the system, and, in turn, makes the construction of (minimal) cover simpler.

The significance of the proposed approach is many-fold. For example, a system designer would preferably work for a compatibility relation R on a given S which gives rise to a family of minimum number of MCCs that covers S .

For an illustration of the above algorithm, let R be a compatibility relation on a set $S = \{x_1, x_2, \dots, x_5\}$ defined by the compatibility matrix (CM) as follows (note that the elements of a CMCC may be PC or NPC):

	x_1 x_2 x_3 x_4 x_5	CMCCs	CCs	MCCs
x_1	1 1 0 1 0	$\{x_1, x_2, x_4\}$ (PC)	$\{x_1, x_2, x_4\}$	$\{x_1, x_2, x_4\}$
x_2	1 1 1 1 1	$\{x_1, x_2, x_3, x_4, x_5\}$ (NPC)	$\{x_2, x_4, x_5\}$	$\{x_2, x_4, x_5\}$
x_3	0 1 1 0 1	$\{x_2, x_3, x_5\}$ (PC)	$\{x_2, x_3, x_5\}$	$\{x_2, x_3, x_5\}$
x_4	1 1 0 1 1	$\{x_1, x_2, x_4, x_5\}$ (NPC)	–	–
x_5	0 1 1 1 1	$\{x_2, x_3, x_4, x_5\}$ (NPC)	–	–

IV. APPLICATION OF MAXIMAL COMPATIBLE SETS TO NETWORK SEGMENTATION

In a network system, issues pertaining to network congestion are encountered due to heavy traffic on the network

or increase in the number of users sharing it. This can reasonably slow down the performance of the network [10].

More often than not, more data are required to be added to a shared network. As a result, performance deteriorates due to competition for *Ethernet bus* by users (competitors) of the shared network [3].

In the event of problems occurring with too many nodes in the same collision domain, suitable network segmentation is introduced to reduce collision, retransmission and contention for bandwidth [3]. Usually, in a segmented network, most of the *communication activities* are performed by the subnets. At times, the sub-network gets flooded with messages reporting *status* and *throughput statistics* to the *Network Monitoring Centre (NMC) programs*, and as users' traffic increases, larger network capacity is consumed. Not very frequently, lack of provision for coordinating and analyzing performance measures at terminals in the sub-network as well as at the host tell upon the efficiency of the network segmentation method (NSM).

Network segmentation is an act of splitting a computer network into *subnets* comprehended as *network segments* or *network layers*. A segmentation of a congested network consists in its decomposition into smaller segments which give rise to a sort of partition. Essentially, a physically separate path (not intersecting with another path) for each pair of communicating ends needs to be built, which tells upon efficiency as well as costing.

We propose that the notion of *covering* instead of *partitioning* a congested network may be applied to obtain a more compact and cost effective network. Thus, in order to achieve a competing segmentation of a congested network, a suitable *compatibility relation*, instead of an *equivalence relation*, needs to be defined. This, in turn, gives rise to a decomposition of the network into (maximal) compatibility classes (segments) whose union is its minimal covering. It is observed that this procedure does not disconnect any segment or sub-network from the entire network, since the pairwise intersection of these (maximal) subnets is non-empty.

Essentially, a compatibility relation based segmentation procedure segments a congested network into subnets (maximal compatibility blocks). This, in turn, structures the network in such a way that no node of any given subnet communicates to every node of another subnet. As a result, the communicating network is reasonably minimized without adversely affecting its goal. In fact, when the network is segmented, there is a tradeoff between time and quality data. That is, if a particular node x_i in a subnet A does not relate to a node x_j in subnet B ,

but needs information from node x_j , then time is traded off to obtain that information from x_k , whenever x_i relates to node x_k in B . This is because the rate of packet delivery from x_j (source) to x_i (destination) is equal to the rate of packet delivery from x_k to x_i plus some time $t > 0$. However, there is a compensation for this delay. The compensation being that the maximal compatibility class (maximal subnet) through this node x_j (source) lowers its throughput to meet that of a (possible) congested path. This type of congestion control is similar to *rate-based technique*.

Further, it is observed that x_k above is a link from subnet A to subnet B . Therefore, such x_k in any subnet supports high bandwidth and low latency (the amount of time taken for a packet to travel from source to destination) between subnets A and B , since x_k can be reached.

A schematic representation of how a *CR-based network congestion control* procedure works:

Let N be a network system on which a compatibility relation is defined with compatibility classes C_k ; $k = 1, 2, \dots$.

Label x_k as L_k (links) if for some compatibility classes C_i and C_j , we have $C_i \cap C_j = \{x_k\}$, for some i, j, k .

These L_k 's in N form its backbone.

Now, the packets are routed through the network. It is important to observe that the performance measure of a refined network will be a function of the cardinality of the set of L_k 's.

Further, there exists at least one node in C_i which maintains a separate queue for any other subnet C_j , and a separate queue for nodes in itself.

Denote the first queue as q_1 and the second queue as q_2 . Then any node within a compatibility class (subnet) C_i maintains a separate q_2 for each L_k in compatibility set C_i and maintains q_1 for nodes in any other compatibility class C_j . Also, it follows that each $L_i \in N$ maintains q_2 for some L_j ($i \neq j$) in N . Therefore, communication flow can be re-channelized and assigned to the appropriate L_k 's with the information about the network traffic and topology. In fact, accounting for the instantaneous queue length between L_k 's, controller (which implements the electronic circuitry required to communicate using a specific physical layer and data link layer standard, such

as Ethernet, Wi-Fi, etc.) allocates packets to the least congested link.

The following diagram illustrates the concept described above.

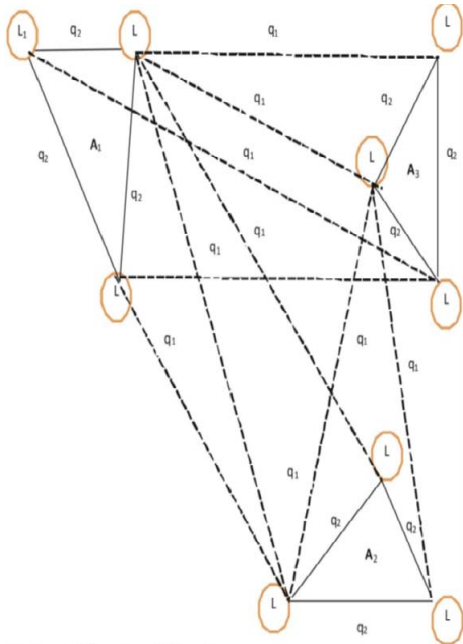


Fig.2:

The diagram (fig. 2) represents the links and queues.

V. APPLICATION OF MAXIMAL COMPATIBLES TO NETWORK DECENTRALIZATION

In a *decentralized* network system, issues pertaining to distribution of *control* versus optimum *communication flow* have been drawing attention since long: *manageability* becomes constrained as the system grows larger ([9], [23]).

A decentralized network is a scheme that allocates resources (both hardware and software) to each individual work station. It aims at connecting users and resources in a *transparent*, *open*, and *scalable* manner. This is achieved by creating multiple locations that support different operating systems in order to avoid occurrences of a complete neutralization caused by any form of attack

([22] provides a diagrammatic description of a decentralized network).

Usually, in a decentralized network, no node has complete information about the states. A node makes decision based only on local information it has. This, in turn, makes diagnosing problems somewhat difficult in the event whereby troubleshooting analysis requires inspecting communications between nodes. Moreover, the *far-flung* distribution of control tends to make decentralized network difficult to manage.

It is possible that two nodes from different ends on the network could try to send data to one another. If no direct link exists between these nodes, the time taken to send one byte of data will depend on the distance between source and destination nodes. Therefore, in a decentralized network, *wire delay* is a bottleneck. Some disadvantages of a decentralized network are mentioned in [22].

In a decentralized network, different segments may control and maintain their various *activities (work)*. It consists of configured systems such that multiple nodes (servers) in the network provide the services used by other nodes (clients) of the network. A typical example of a decentralized network is one in which nodes (clients) utilize different nodes (servers) to carry out their various *functions* ([13] provides many examples) of decentralized network.

Schematically, the graph of a compatibility relation \mathbb{R} of fig. 1 may be viewed as a decentralized network if, each node is a client and a server as well. It is observed that the symmetric property of a compatibility relation guarantees each node in the graph of \mathbb{R} to assume *client – server functions*. Since the topology of \mathbb{R} , viewed as a decentralized network, is identical to a mesh topology, there is reasonably enhanced *reliability* and *speed* in a decentralized network. Also, it is observed that this advantage grows significantly when the decentralized network is configured with *redundant* servers (nodes) - multiple nodes providing the same service.

Thus, by applying the notion of MCCs, the distribution of control to nodes would be reasonably minimized without adversely affecting the goal of network decentralization.

VI. CONCLUDING REMARKS

Concluding, it can be emphasized that in a foreseeable future, besides in *hard sciences* like Engineering, Computer sciences, etc., in order to model problems in *soft sciences* like biology, economics, sociology, etc., which are invariably supposed to deal with a large number

of incompletely specified interactions, applications of compatibility relation would play an important role.

REFERENCES

1. Biswas N. N., "Maximum Compatible Classes from Compatibility Matrices", *Sadhana*, 14 (1989), 213 – 218.
2. Caldwell SH., "Switching Circuits and Logic Design", John Wiley, 1958.
3. Cannon K and K. Caudley, "CCNA Guide to Cisco Networking", third edition, 4(10) (2007).
4. Charkraborty A., "On State Minimization of Very Large Incompletely Specified Sequential Machines", BE Project Report, Dept. of Electrical Communication Eng. Indian Inst. Of Sci., 1987.
5. Grasselli A., "A Note on the Derivation of Maximal Compatibility Classes", *Calcolo*, vol. 3 (1966), 165- 176.
6. Grasselli A and F Luccio, "A Method for Minimizing the number of internal states in Incompletely Specified Sequential Networks", *IEEE Trans. on Electronic Computers*, 14 (1965), 350- 359.
7. Grasselli A., "Control units for Sequencing Complex Asynchronous Operations", *IRE Trans. on Electronic Computers*, 11 (1962), 483- 493.
8. Hill F.J and G R Peterson., "Introduction to Switching Theory and Logic Design", John Wiley, 1968.
9. Jacob A and C. Bullen, "Centralization versus Decentralization of Information Systems": A critical survey and an annotated bibliography, Centre for Information Systems Research, 11 (1978), 112- 114.
10. Kelvin L.M, "Performance Measurement Problems in a Packet-Switch Network", *System Development Corporation*, 5(1981), 29-40
11. Kohavi Z., "Switching and Finite Automata Theory", 2nd edition (1978) McGraw- Hill.
12. Kurepa DJ., "On Reflexive Symmetric Relations and Graphs", *RazpraveDissertationes, SlovenskaAkademija in Umetosti*, 1952 – 1953, pp.1- 28.
13. Lau R, Persiano, R, and P Varaiya., "Decentralized information and control": A network flow example, *Automatic Control, IEEE Transactions*, vol. 17 (1972), 466- 473.
14. Lipovski S and M.P Marcus., "Tolerance Rough Sets, Cech Topologies", *Learning process, Bull. Polish Academy of Science, Technical Science*, 42, 3(1994), 471 – 487.
15. McCluskey EJ., "Minimum- State Sequential Circuit for Restricted Class of Incompletely Specified Flow Tables", *Int. Symposium on Theory of Switching Systems and Finite Automata*, 10 (1962), 1759- 1767.
16. Paull M.C and S.H Unger., "Minimizing the number of states in Incompletely Specified Sequential Switching functions", *IRE Trans. on Electronic Computers*, 8 (1959), 356- 367.
17. Pognowski J., "Tolerance Space with Applications in Linguistics". Pozhan Univ. Press, Pozhan, 1981.
18. Singh D, Dunari, M.H and J N Singh., "A Note on Covering and Partition of a Finite Set", *Math. Sci. Math. Edu.* 5(2) (2010), 1- 5.
19. Singh D and T. O William-west., "A Note on Application of Compatibility Relation to Switched Network", *Microwave Int. Journal of Sci. and Tech.* 3(1) (2012), 56- 62.
20. Tremblay J.P and R Manohar., "Discrete Mathematical Structures with Applications to Computer Science", McGraw- Hill, 1975.
21. Weissman J., "Boolean Algebra, Map Coloring and Interconnections", *American Math. Monthly*, vol. 69, 608- 614.
22. Wikipedia: user contributed 11(2012).
23. William C.W.M., "On the Minimal Set of Compatibles for Closure", *Microwave research Institute*, 1962, pp.1- 46.