ON DECISION PROCEDURE

KANTILAL DAS

Like many scientific theories logical theories attempt to design instruments for measuring logical properties. Unlike theoretical investigation the nature of logical tests and procedures is peculiar, since they are purely mechanical which makes the all important distinction between logical theories and other theories. Logical theories, by and large, can be carried out by means of mechanics. This paper will attempt to survey a very important device which is simply called the device of decision procedure.

The whole course of this paper will be studied in three sections. In section one some theoretical aspects of the concept of decision procedure will be discussed. Section two deals with the examination of some logical systems which are not considered as decision procedures, and section three will be devoted to account some other logical systems which are considered as decision procedures.

I

The concept of decision procedure is predominantly concerned with the concept of decidability. Every logical system, by and large, requires a recipe for deciding whether there is some effective or mechanical procedure by which we can tell a *wff* either as valid or not; an inference either as sound or not. Such a procedure is called a decision procedure. It is usually laid down that a decision procedure must be effective in the sense that it is precisely stable, reliable and finite "which can be described in advance for providing in a finite number of steps a "yes" or "no" answer to any class of questions."[1] It is mechanical, since it acts like a computing machine so effectively that with respect to any formula it would at some point give a positive answer if the formula under consideration were in fact valid or a negative answer if the formula under consideration were in fact not valid.

Indian Philosophical Quarterly, Vol. XXV No. 3
July 1998

Decision Problem :

To examine whether a formula is valid or not is a matter of decision and it has been effectively decided by a decision procedure. The problem of discovering a decision procedure is called the *decision problem.* "*The decision problem* for any deductive system" as Copi says, "is the problem of stating an effective criterion for deciding whether or not any statement or well formed formula is a theorem of the system."[2] This means that so far as the logical systems are concerned we have so many different types of decision problems. The problem of discovering a decision procedure for a predicate is called the decision problem for that predicate. Analogously, the problem of finding out a decision procedure for a class is called the decision problem for that class. So every class of objects has a decision problem; i.e., the problem of finding out an effective method for deciding whether it is a member of that class or not. There we find two types of decision problems, such as solvable and unsolvable. A decision problem may be solvable for some classes or may be unsolvable for some others. A decision problem is slovable if a decision procedure can be found within the system; unsolvable if a decision procedure can not be found within the system. If the decision problem can be found within the system, the system is often said to be decidable; if a decision procedure can not be found within the system, the system is said to be undecidable. Hughes and Londey have observed that every logical system possesses a decision problem, but not every logical system is a decidable or solvable decision problem.[3] So the decision problem for any system is to find out an effective method so as to decide arbitrarily whether any *wff* of propositional calculas (pc) is valid or not. If a decision procedure is found in the predicate, the predicate is said to be effectively decidable; if a decision procedure is found in the class, the class is said to be effectively decidable. So far as the decision problem is concerned any theory or any *wff* can be answered in the affirmative if it is decidable or it can be answered in the negative if it is undecidable. An example of a decidable theory is the statement or propositional calculas, since a formula is a theorem iff it is a tautology. The method of truth-tables provides a decision procedure, since it constitutes a solution to the decision problem and it enables us to decide effectively whether or not any *wff* is a tautology.

Does a Decision Procedure Make Other Rules Obsolete?

If it is admitted that a decision procedure is an effective method which allows us to decide whether a given *wff* is valid or not, then a very natural question arises at this juncture. The question is : whether a decision procedure is responsible or not responsible for making other rules obsolete. As we have a purely mechanical or computing system at our hand for testing the soundness of an inference, why should we continue to use the rules at all? Several points may be given in response to this question. The first point is that although truth-value assignments may tell us that a certain conclusion can be drawn from certain premises, they do not tell us how to draw the conclusion. A computer or a machine gets its action if it is being operated by following certain rules. Similarly, a decision procedure may be shown to be effective if it follows in accordance with rules. Only the rules can tell us, not the method or process, how one statement can be deduced from another. Logic, of course, may be equated with mathematical reasoning, but this does not mean that logic can be formulated exclusively on the basis of matrices and truth-value assignments. We do apply matrics and truth-value assignments only for testing mathematical or logical reasoning; but does it mean that they stand for mathematics and logic? Certainly not. A conclusion is derived from the premise/premises not in terms of truth-values, but in terms of logical rules. Finally and importantly, 'even if a satisfactory theory of propositional deduction could be formulated in terms of a decision procedure,' as Anderson and Johnstone Jr. have observed, 'such a theory could not be formulated for certain more complex kinds of reasoning for which it is known that there cannot be a decision procedure.'[4] So what we can say about it is that a decision procedure does not ignore rules; but it acts in accordance with rules.

II

We have already pointed out that in logic we have some systems which are not decidable; i.e., in these systems we do not find any decision procedure. In this section we examine some logical systems which are not decision procedures.

Proof is not a Decision Procedure :

Proof procedure is not a decision procedure; since in the case of a proof, there remains nothing to be proved as false. If we are asked to prove a formula, we know prior to its proving that it is logically true; but we do not know that it is basically false. This indicates the difference between 'proving something' and 'testing something'. In order to prove something, we have no option to make it false. But in order to test something the given may be either true or false and it is the doer's task to decide effectively whether the given is true or false. Proof, thus, establishes logical truth; but not logical falsity. But Quine observes that the impossibility of a decision procedure for validity or proving something does not prevent us from developing procedures for proving validity. He, however, points out the difference between a decision procedure and a proof procedure in saying that "a decision procedure assures an affirmative and negative answer every time, while a proof procedure assures at best an eventual affirmative answer where an affirmative answer is in order."[5] A complete proof procedure differs from a decision procedure, says Quine, for a complete decision procedure enables us to give both positive and negative answers; but a proof procedure gives only a positive answer, 'it does not deliver negative answers.'[6]

If we adhere to the principle that a proof procedure is not a decision procedure as it fails to give a no answer to a formula, as Quine says, then what do we think about *Conditional Proof (CP) and Indirect Proof (IP)* ? The rule of CP can be used in dealing only with valid arguments, it can not be a decision procedure.

Like CP, IP is also a proof procedure and it should not be treated as a decision procedure. But a little bit of confusion may arise here if it has been admitted that IP is equated with RAA *(Reductio Ad Absurdum).* RAA method is considered, we shall examine later on, as a decision procedure.[7] The question is : If RAA method is considered as a decision procedure and IP is often called the method of RAA[8], then why IP is not considered as a decision procedure? In what sense IP is often called RAA? The similarity is that like RAA, IP also attempts to have a contradiction of the given or to reduce an absurdity of the given; but the way through which it is drawn is different. In RAA method the absurdity is drawn in terms of truth-values; but in IP the absurdity is drawn in terms of proposition or propositional variables. If the given statement is valid then its negation is a contradiction (*both in RAA and IP*); and hence the given has to be a tautology. But if the given statement is not valid, then its negation does not lead into a contradiction and it is not to be proved as a tautology. In such a case the given statement must be either contingent or inconsistent. So far as the truth-values are concerned, RAA method enables to decide effectively both inconsistent and contingent statements. But as far as the inferential rules are concerned IP fails to determine whether a given statement is contingent or inconsistent. As a proof procedure, IP gives only a positive answer to any statement; but there is no scope of finding out a negative answer in IP and in this sense IP should not be considered as a decision procedure.

One Variable Quantification System (OVQS):

OVQS is not a decision procedure as it possesses an infinite number of interpretations. It is often claimed that there is no mechanical procedure which is so effective in determining validity, implication etc.,in OVQS. Every logical system, of course, has possessed some mechanical device, but not every logical system has an *effective* mechanical device. Accordingly, OVQS has rules, but the rules it possesses have wide range of application. It has possessed some mechanical device, but the mechanism is not effective like truth-table system, since the mechanism of OVQS is more complicated to be introduced and in most cases it is not so efficient as the means as we shall use in truth-table system.

It is true that the rules of derivations that we have in OVQS are sound and complete as they enable us to prove an argument in OVQS. But the technique through which we attempt to prove a schema in OVQS is not purely or effectively mechanical as it requires some insight and ingenuity on our own part. Here we are trying very hard to prove a particular schema but we fail to do so. It is very often to be the case, because there we have so many interpretations and we may fail to choose the right one to prove the schema as valid. If we fail to make a schema as valid, then does it mean that the schema is invalid? It does not. It may be the case that there is a proof which we fail to discover by applying the correct rules or correct interpretation. So it is up to the doer to decide which interpretation is effective for making a schema as valid. But if we attempt to prove a formula in truth-table system and eventually fail to prove it, then, of course, we can always check the truth-table system for validity, but we can not do this for those OVQS. Here we have to be careful of the problem of UI,EI, UG, EG. and their mutual transformations into each other. This opens many interpretations of this system. So it is claimed that "our rules of derivation do not furnish a decision procedure for the validiy of OVQS. What they *do* furnish is called a *proof procedure*, that is, a method which will produce a proof of a schema if it is valid, but which will not tell us that the schema is invalid *if* it happens to be"[9]. In logic there we have some schemata which are neither valid not inconsistent. OVQS as a proof procedure fails to prove this formula. That is why Resnik aptly puts," ..... there are also schemata which are neither valid nor inconsistent, and our proof procedure will not establish the consistency or invalidity of these schemata; nor of the negations."[10] It mens that OVQS does not yield a decision procedure. Resnik says, "The moral is that one-variable quantification theory is going to place much greater demands upon our ingenuity than truth-functional theory did."

Quantificational Validity is not a Decision Procedure :

Like OVQS, there is no effective mechanical procedure for quantificational validity. That is to say that we do not find here any effective method applied by man or a machine to decide arbitrarily whether a quantificational theory is valid or not. This had been developed by Alonzo Church in claiming that logic can never be fully mechanised. After Church, it was Gödel who had established a remarkable incompleteness theorem in comparing mathematical truth with provability.

But again this does not mean that quantificational logic lacks rules of inference. Quantificational logic can definitely be endowed with sound inferential rules. The rules of inference that we have in quantificational logic are, basically, the same as the rules of those used in OVQS. Furthermore, truth - functional rules may be reused here without any change. Here we find some more extentions and restrictions of instantiations and generalisations of universal and particular propositions. But the intuitive basis of these rules remains the same, since no further rules are necessary. So our all important conclusion is that although we can not test the validity, equivalence, etc. in quantificational logic like many other systems we can take the advantage of these rules to establish the presence of these properties and relations. But quantificational logic is not still a decision procedure, because to be sure of anything about in this system we have to depend "somewhat on luck and ingenuity"[12] on our own part.

Tree-method is not a Decision Procedure :

Tree-method is not a decision procedure. Although it is true that tree-method adequately formalizes logic, the method is claimed to be inadequate. It is inadequate, because in order to test an invalid inference in tree-method, one may go on for ever. There we do not find any terminus point or any finite number of steps after which the machine classifies a given invalid inference as invalid.

Let us examine the following invalid inference :

1.	(x) (∃y) x Ly	1 ∴ aLa.
2.	- aLa	[denial of the conclusion]
✓3	(∃y) aLy	[1. by (x)]
4.	aLb	[3 by (∃y)]
✓5.	(∃y) bLy	[1 by (x)]
6.	bLc	[5 by (∃y)]

.

.

.

Invalid

The above inference is invalid. In answering to the question "Is this inference valid?" the machine fails to give a 'yes' or a 'no' answer since the machine could not finish the tree. The premise of the above inference starts with a universal quantifier following an existential quantifier. Hence step no-I can never be cheeked off, because in this step the universal quantifier i.e., '(x)' can be legitimately instatiated by any individual constant whatsoever. So the option of instatiation in this step remains open. Now, once we instantiated the universal quantification by any individual constant, say a, then the subsequent instantiation of any existential quantification can not be legitimately applied on the same individual constant. This means that every instantiation of existential quantification brings a new individual-constant and this process is continuing for ever. Thus, so far as the procedure is concerned, tree-method may remain open for ever in the case of an invalid inference. But if at some point the machine could predict, of course, intuitive prediction, it will never finish the tree; it could tell us at that point that the correct answer is 'no'. That is why Jeffrey aptly remarks that in tree-method. "We have an adequate "yes" machine which is inadequate as a "no" machine."[13]

Tree-method is inadequate not only for quantificational validity; but also it is inadequate for testing consistency and inconsistency. This is mainly for the reason that here we are adding individual constants in every existential instantiation and this process always opens new path. Let us examine the following set of sentences for testing consistency or inconsistency.

$(x)\ [Nx \supset (\exists y)\ (Ny\ .\ Rxy)].$

$(x)\ \{Nx \supset (y)\ \{Ny \supset (2)\ \{Nz \supset [(Rxy.Ryz) \supset Rxz]\}\}\}$

$(x)\ (Nx \supset -\ Rxx)$

Na

✓ $Na \supset (\exists y)\ (Ny.Ray)$

$-Na$ ✗   ✓ $(\exists y)\ (Ny.Ray)$

$Nb.\ Rab$

Nb

Rab

✓ $Nb \supset (\exists y)(Ny.Rby)$

$-Nb$ ✗   ✓ $(\exists y)(Ny.Rby)$

✓ $Nc.Rbc$

Nc

Rbc

✓ $Na \supset (y)\ \{Ny \supset (z)\ \{Nz \supset [(Ray.Ryz) \supset Raz]\}\}$

$-Na$ ✗ $(y)\ \{Ny \supset (z)\ \{Nz \supset [(Ray.Ryz) \supset Raz]\}\}$

Nb ⊃ (z){Nz ⊃ [(Rab.Rbz) ⊃ Raz]}

-Nb (z){Nz ⊃ [(Rab.Rbz) ⊃ Raz]}

Nc ⊃ [(Rab.Rbc) ⊃ Rac]

-Nc (Rab.Rbc) ⊃ Rac

Na ⊃ -Raa

-Na -Raa

-(Rab.Rbc) Rac

-Rab - Rbc

Consistent.

The initial set consists of three universal sentences which can never be checked off. After the instantiation of the first sentence of the initial set we get a new existential sentence. Every instantiation of a new existential sentence gives birth to a new individual constant, since no existential sentence can be legitimately instantiated with the same individual constant which has already been occurred in the previous step. In the above procedure the existential sentence '(∃y) (Ny.Ray)' gives birth to a new individual constant such as b; and that leads to '(∃y) (Ny.Rby)'. And it again gives rise to another individual constant, say c; and that leads to (∃y) (Ny.Rcy). But how long will this process continue? It will continue for ever.So we can never construct a model based on just a finite number of individuals. So we can say that tree-method can not be a complete decision procedure, because at least in some cases there is no finite number of steps and no finite path yields a successful model. Not all paths close, but the process never ends.

III

In this section we shall examine some logical methods which are considered as decision procedures.

Truth-table Method is a Decision Procedure :

Truth-table method is an ideal example of a decision procedure. It is intuitively clear that the decision problem of PC is solvable in truth-tables method, since it enables us to decide effectively the validity of any *wff.* Every *wff* of PC has a truth-table, because every *wff.* of PC is builtup entirely from truth-functions, and every truth-function has a truth-table. To introduce this method we first consider how one can construct a truth-table for any *wff.* of the sentential language. Suppose we construct a truth-table for this formula :

$$(p.-q) \supset (p \vee -q)$$

A truth-table for a formula is a table in which the possible combinations of truth-value for the component variables of the formula are listed and the resulting truth-value of the formula for each possible combination indicated. The formula under consideration is made up of two variables, such as p and q. This means that there are four possible-ways in which truth-value may be assigned to the statement variables which may be set down as follows :

P	Q
T	T
T	F
F	T
F	F

If the formula under consideration is made up of three variables, it has eight possible ways, if it is made up of four variables, it has sixteen possible ways. Thus truth-table procedure satisfies 2^n principle; i.e., in truth-table one variable means 2^1 or two rows; two variables means 2^2 or four rows; three variables mean 2^3 or eight rows; four variables means 2^4 or sixteen rows and so on so forth.

Now the truth-table for the given formula is constructed as follows :

p	q	(p-q) ⊃ (pv-q)	
T	T	F (T) T	
T	F	T (T) T	
F	T	F (T) F	
F	F	F (T) T.	Tautology

It seems clear from the above that truth-table method is so effective that with the help of this machine we can decide effectively the truth-value of all PC. But this procedure may be onerous or embarrassing if we have more then three vaiables in a wff. The problem may be cumbersome; but still the method remains to be effective.

Tautology, Inconsistent and Contingent :

So far as the values are concerned, logical statements can be classified into tautology, inconsistent and contingent. A statement is tautology if it is necessarily true; inconsistent if it is necessarily false; and contingent if it is neither necessarily true nor necessarily false. The statement : "Either India is a democratic country or not a democratic country" is a necessarily true statement. The statement "India is both a democratic country and not a democratic country" is a necessarily false statement; the statement: India is a democratic Country" is a contingent statement, since its truth or falsity depends on the existing states of affairs. It is important to note that the negation of a tautology leads into an inconsistency; the negation of an inconsistency leads into a tautology and the negation of a contingent leads to a contingent. '(pv-p)' is necessarily true and hence tautology and its negation (p.-p) is necessarily false and hence inconsistent; and vice-versa. But '(p ⊃ q)' is a contingent statement and its negation; i.e., '-(p ⊃ q)' is also a contingent statement. Truth-table method is effective in determining whether a statement is tautology, inconsistent or contingent.

Reductio Ad Absurdum Method is a Decision Procedure :

Reductio Ad Absurdum (RAA) method is considered to be a decision procedure, because all truth-table methods are decision procedures and RAA method is a short-cut truth-table method, so it follows logically that RAA method is a decision procedure. It is one of the many short-cut truth-table methods. In this method the truth-value of a given formula can be decided very quickly and very effectively than any other truth-table method. That is why Copi aptly says, "The *reductio ad absurdum* method of assigning truth-values is by far the quickest and easiest method of testing arguments and classifying statements"[14]. Let us explain the method of RAA.

The aim of this method is to prove something indirectly and it tries to get the contradiction of the given. Suppose, A is a *wff* and let us further suppose that A is not a tautology and we express this supposition by writing F under the main operator of A. Now, if we go ahead with the supposition that A is not a tautology and proceed to fill in truth-values accordingly for the components of A, we ultimately lead into either contradictory truth-values, if the given statement were in fact a tautology, or we can consistently assign truth-value to each of the components if the given statement were in fact not a tautology. If we arrive at a contradictory truth-values of any one of the components of the denial of the original statement, then our earlier supposition that 'A is not a tautology' happens to be false and as a matter of fact the given statement is proved to be a tautology. But if we consistently assign a truth-value of the components, then A can not be a tautology. Copi says, "If it is possible to assign truth-values consistently to its components on the assumption that it is false, then the expression in question is not a tautology, but must be either contradictory or contingent"[15]. In such a case we make an attempt to make it true by assigning truth-values to its components. If the assigning truth-values leads into a contradiction, then the formula must be inconsistent; but if the assigning truth-values make it true in some cases and false in others the formula happenes to be a contingent. This method is called the method of *Reductio Ad Absurdum* method.

We have pointed out earlier that the negation of a tautology leads into a contradiction. RAA method admits exactly the same principle, since if the given statement under consideration is a tautology then its negation in question would be a contradictory. But if the given statement is not a tautology, say, a contingent, then its negation would be a contingent too.So the technique of RAA can be summed up in the following way :

(i) Assume first that the given formula is not a tautology by placing F under the main connective of the given formula.

(ii) Follow up the two-fold consequences : Either we finally arrive at a contradiction or we do not arrivle at a contradiction.

(iii) If we finally arrive at a contradictory truth-value of any one of the components of the given, the formula is a tautology; if we do not have, the formula is not a tautology.

(iv) Now, if the formula is not a tautology, then it must be either a contingent or an inconsistent formula. In such a case we assign truth-values to make it true instead of false.

(v) If this attempt leads to a contradiction; the formula is a contingent.

Let us test the following concrete example :

(i) $(A \supset B) \supset [(B \supset C) \supset (A \supset C)]$

Let us assume that the given statement is not a tautology by placing F under the main connective and we get the following :

(ii) $(A \supset B) \supset [(B \supset C) \supset (A \supset C)]$

F

Now, ((ii) can be false if its antecedent is to be true and its consequent is to be false and we have :

(iii) $(A \supset B) \supset [(B \supset C) \supset (A \supset C)]$

T F F

Now, to make '$[(B \supset C) \supset (A \supset C)]$' to be false, we have to assign T to its antecedent and F to its consequent. Thus we have :

(iv) $(A \supset B) \supset [(B \supset C) \supset (A \supset C)]$

T F T F F

Now, '$A \supset C$' is false, if A is true and C is false. And if C is false and to make '$B \supset C$' as true; we have to assert B is false. And, again if B is false and to make '$A \supset B$ as true, we have to assert A is false.

So we have the final step like this :

(v) $(A \supset B) \supset [(B \supset C) \supset (A \supset C)]$

F T F F   F T F F   T F F

It seems clear that in order to make the statement as false, we are forced to assign the truth-value of A both F and T. But this is false and it leads into a contradiction (assuming Peirce's Law that false leads to a contradiction)[16]. So our assumption that the given statement is not a tautology is false, and it sends us back to admit that the given statement is a tautology. Likewise are can effectively decide contradictory and contingent statements by applying RAA method. Hence it is called a decision procedure. Copi says, ".... the reductio ad absurdum method is superior to any other method known"[17].

Conjunctive Normal Form (CNF) is a Decision Procedure :

Although the validity testing of all types of normal forms (such as CNF, DNF, BCNF, BDNF, PCNF, PDNF) is comparatively laborious than RAA and other decision procedures, it is still considered as a decision procedure and its utility is highly important for theoretical purpose in logic. For the sake of brevity of this paper, we restrict our discussion only to CNF.

Hughes and Cresswell have defined CNF like this : "A *wff.* is said to be in *conjunctive normal form* (CNF) if it is a conjunction (possibly degenerate), each conjunct which is a disjunction (again possibly degenerate), the disjuncts which are of certain specified forms"[18]. In another book, Hughes along with Londey have said that a wff. is in CNF iff "it is a conjunction of the disjunctions, and no negation sign has an argument other than a single propositional variable"[19]. Keeping the above definitions of CNF in mind any proposition can be put into CNF by following the steps in order :

(i) Apart from '-', 'v', '.', no other logical connectives can be retained in the formula.

(ii) Use only Impl. and Equivalence rules to obtain a formula containing only the required connectives.

(iii) Use De.M. and D.N. to remove all negations outside parenthesis if any.

(iv) Use D.N. to remove all double negations if any and ensure that no variable is preceded by more than one negation sign.

(v) Use the distributive law; i.e., [pv(q-r)] = [(pvq).(pvr)]

as many times as necessary to produce a conjunction of the disjunctions of single formulas.

Let us solve the following formula in CNF :

(A ⊃ B) ⊃ [(B ⊃ C) ⊃ (A ⊃ C)]

≡ -(-AvB) v [-(-BvC) v (-AvC)] [Def. ⊃].

≡ (A.-B) v [(B.-C) v (-A v C)] [De.M., D.N.].

≡ (A.-B) v [-A v C v B) . (-A v C v-C)] [Dist.].

≡ [(A.-B) v (-A v C v B)] . [(A.-B) v (-A v C v - C)] [Dist.].

≡ [(-A v C v B v A) . (-A v C v B v - B)] .
[(-A v C v - C v A) . (-A v C v - C v - B)] [Dist.].

≡ (- A v C v B v A) . (-A v C v B v - B) .
(- A v C v - C v A) . (- A v C v - C v - B) [Dropping brackets].

≡ (.A v - A v B v C) . (- A v B v - B v C) . (A v- A v C v - C)
. (- A v - B v C v - C)[Reordering).

Tautology in CNF

The last line of the above derivation is a conjunction of the disjunctions of single formulae and in this step all negation signs negate single variables accordingly. The formula is a tautology in CNF, because each of the components of the formula is in the form of '(P V-P)'. The conjuncts are disjunction, and a disjunction is valid iff a variable and its negation appears as disjuncts. This is exactly the same that happened in the above CNF. So it is proved as tautology in CNF.

But what do we think about an elementary or a singular formula? Elementary formulae are called degenerate wffs. and they are also to be counted in CNF by following the moral that "Every wff. is a CNF'. "A wff. is a degenerate conjunction" for Hughes and Londey, "if it could appear as one conjunct...."[20]. For example, the wff. 'pv-pvq' is a degenerate conjunction. It is one conjunct of the disjunctions and hence is a CNF. Similarly, p, - p, q, etc., are also to be counted as CNF. 'p', for example, is a degenerate conjunction of degenerate disjunctions. Since "p = (p . p)"(degenerate conjunction) and "p ≡ (p v p)" (degenerate disjunction)(see PC No : 8 and 9 respectively)[21]. To obtain a degenerate conjunction in CNF we have to be careful about redundant variables. Hughes and Londey prescribed the following suggestions :

(i) No redundant disjuncts occur in any conjunct in a CNF. This means that 'p v p' is replaced by 'p' and likewise '-p v - p' is replaced by' -p'.

(ii) No redundant conjuncts occur in CNF. For example; '(p v- p).(p v- p)' is replaced by '(p v - p)'.

(iii) Disjunctions variable appear in the alphabetical order; e.g., (p v - r v q v - p) is replaced by' (p v - p v q v - r).

Thus CNF enables us to decide effectively every wff. of PC either as a tautology or not as a tautology; and hence a decision procedure.

Method of Existential Conditionals is a Decision Procedure :

We have ignored the possibility of decision procedure in OVQS or MPL (Monadic Predicate Logic), because to select a schema of MPL at random and test if for validity is usually sufficient to convince anyone that the procedure is long and tedious, and apart from rules it requires some skill and ingenuity on our own part. But it was Quine[22] who along with von Wright finally[23] overcame the defects of MPL by developing the procedure given by Behmann. Here we examine the. *Method of Existential Conditionals* that has been presented by Quine. His method is to conjoin the premises of an argument and to combine them with the conclusion, using ' ⊃ ' and finally to test the result for satisfiability by the method. In many cases this can be done reasonably briefly.

Let us examine the following argument :

All of the witness who hold stock in the firm are employees. All of the witness are employees or hold stock in the firm. Therefore all of the witness are employees.

Now let : Wx : x is a witness.

Hx : x is a person who holds stock in the firm.

Ex : x is an employee.

By following the above notations we can symbolise the given argument like this :

$(x)\ [(Wx\ .\ Hx) \supset Ex].$

$(x)\ [Wx \supset (Ex \vee Hx)]\ \therefore (x)\ (Wx \supset Ex)$

Now, if we render the above symbolic form of the argument into Boolean statement schemata, we have the following form :

$-\exists WH\bar{E}$

$-\exists W\bar{E}\bar{H}\ \therefore\ -\exists W\bar{E}$

now, we can test the validity of the inclusive Boolean statement schema like this :

$(-\exists WH\bar{E}.\ -\exists W\bar{E}\bar{H}) \supset -\exists W\bar{E}$

$\equiv -(-\exists WH\bar{E}.-\exists W\bar{E}\bar{H}) \vee -\exists W\bar{E}$

$\equiv (\exists WH\bar{E} \vee \exists W\bar{E}\bar{H}) \vee -\exists W\bar{E}$

$\equiv -\exists W\bar{E} \vee (\exists WH\bar{E} \vee \exists W\bar{E}\bar{H})$

$\equiv \exists W\bar{E} \supset (\exists WH\bar{E} \vee \exists W\bar{E}\bar{H})$

$\equiv \exists W\bar{E} \supset \exists(WH\bar{E} \vee W\bar{E}\bar{H})$ [Law of Existential Distribution (LED)]

$\equiv W\bar{E} \supset (WH\bar{E} \vee W\bar{E}\bar{H})$ [By dropping existential Schema].

Now by applying fell-swoop method, $W\bar{E}$ can be true only on one condition, i.e., when W = I and E = O (where I stands for T and O stands for F). Now if the assigning value of the antecedent can be put on the consequent variables and get a truth-value true, then the argument would be proved as valid, otherwise not.

Fell Swoop : $W\bar{E} = I$ if $W = I$ and $E = O$

$W\bar{E} \supset (WH\bar{E} \vee W\bar{E}\bar{H})$

$IHI \vee II\bar{H}$

$H \vee \bar{H}$

(1)

Valid

The outcome of this paper is that every logical system, by and large, has a decision procedure, since every logical system acts in accordance with rules.

But this does not mean that every logical system possesses an *effective* decision procedure. A decision procedure is considered to be effective if it enables us to provide a yes or no answer to a formula with minimum effort. Proof procedure, as we have already noted, is not a decision procedure as it fails to give a no answer to a schema. IP and CP are not decision procedures for the same reason. MPL or OVQS is not effective, because apart from rules it requires human skill and ingenuity for proving validity. Quantificational logic is not a decision procedure since it possesses an infinite number of interpretations, and to choose the right interpretation for testing validity one has to depend again on human skill and ingenuity. Tree-method is not effective too, as it lacks to give a no answer in the case of an invalid argument. But all methods relating to truth-values enable us to provide a yes or no answer to any sort of test. Proof procedure can at best be considered as a qua-decision procedure. Quine says that a proof procedure is only "a half of a decision procedure"[24].

NOTES

1. Stoll, R. R.*Set Theory and Logic;* Eurasia Publishing House; (Pvt); New Delhi; 1967 p. 308.
2. Copi, Irving: *Symbolie Logic;* Macmillan Publishing Co., Inc. New York; 1979, P. 257.
3. Hughes, G.E. and D.G.Londey:*Formal Logic*, B.I Publications; P.56.
4. Anderson,T. M and Henry W.Johnstone, Jr.:*Natural Deduction : The Logical Basis of Axiomatic Systems*; The Pennsylvania State University; Wadsworth Publishing Company, Inc.; 1962. P.85.
5. Quine, W. V.: *Methods of Logic*; Routledge and Kegan Paul, Ltd; 1952, P.142.
6. ....:*Ibid.* P.181.
7. Carney, J. D:*Introduction to Symbolic Logic*; Prentice-Hall, Inc. Englewood Cliffs, New Jarsey; p.45.
8. Copi, Irving : *op. cit.*p.52.
9. Resnik, M. D. : *Elementary Logic*; McGraw-Hill Book Company; New York: 1970, P.225.

10. ...... : *Ibid.*, p. 225.

11. ..... : *Ibid.*, p. 226.

12. ..... : *Ibid.*, p. 321.

13. Jeffrey, R. C. : *Formal Logic : Its Scope and Limits*; McGraw- Hill Book Company, New York; 1967, P.196.

14. Copi Irving : *op.cit.* p.62.

15. ..... ': *Ibid.*, p.62.

16. .... : *Ibid.*, p. 62.

17. .... : *Ibid.*, p. 62.

18. Hughes, G. E. and M. J. Cresswell : *Modal Logic*; Wellington, New Zealand; 1971, P.16.

19. Hughes, G. E. and D. G. Londey : *op. cit.* p.73.

20. ...... : *Ibid.*, pp. 73-74.

21. Hughes, G, E and M. J. Cresswell : *op. cit.*, p.15.

22. Quine, W. V. : "On the Logic of Quantification", *Journal of Symbolic Logic*; vol. 10, 1945, Pp 1-12.

23. Von Wright, G. H. : *Logical Studies*; London; Routledge and Kegan Paul, 1957. pp. 1-43.

24. Quine, W. V. : *Methods of Logic*; p. 182.

306