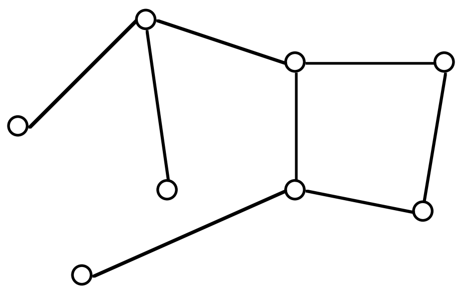
Metalogicon (2000) XIII, 1

Standard Sentence Logic as Graphs

Arturo Graziano Grappone

1. Definitions

To describe graphs we use Bollobás' definitions. Graph G is a pair of disjointed sets (V, E) such that E is a subset of the set of unordered pairs of V. Set V is the set of *vertices* and E is the set of *edges*. A graph has a typical graphic representation.



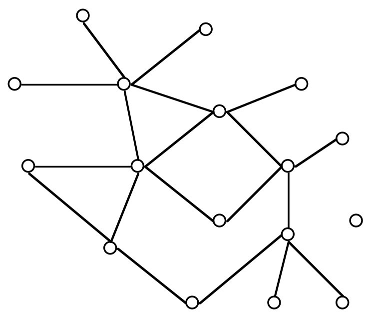
Example of graph: the circles are the vertices and the segments are the edges.

If G is a graph then $V=V(G)$ is the vertex set of G and $E=E(G)$ is the edge set. An edge $\{x, y\}$ is said to join the vertices x and y and is denoted by xy (obviously, $xy=yx$) where x and y are the *endvertices* of xy. If $xy\in E(G)$ then x and y are *adjacent* or *neighbouring* vertices of G and the vertices x and y are *incident* with the edge xy. Two edges are *adjacent* if they have exactly one common endvertex. The set of vertices adjacent to a vertex $x\in G$ is denoted by $\Gamma(x)$. The *degree* of x is $d(x)=|\Gamma(x)|$. The *order* of graph G is the number of its vertices $|G|$. The *size* of graph G is the number of its edges. We can prove that if $|G|=n$ then $e(G) \leq \binom{n}{2}$.

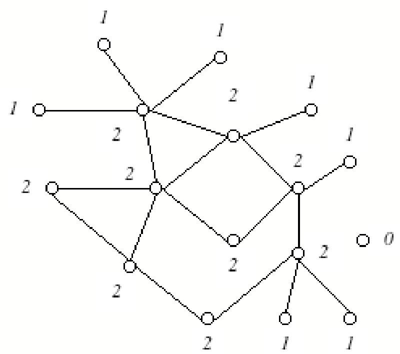
Given any graph $G=(V,E)$ such that $|G|=n$, call *level* of a vertex x $\Lambda(x)$ a natural number which is associated with x: To calculate the levels of the vertices of a graph use the following procedure:

a) Assign $\Lambda(x)=0$ to all the vertices of degree 0, $\Lambda(x)=1$ to all the vertices of the least degree greater than 0, $\Lambda(x)=2$ to the remaining vertices.
b) Increase by n-1 the level of any vertex of the biggest level which has the n-th biggest number of edges with vertices of lesser levels.
c) If there are increments of levels in the last step b) return to b), otherwise stop the procedure.

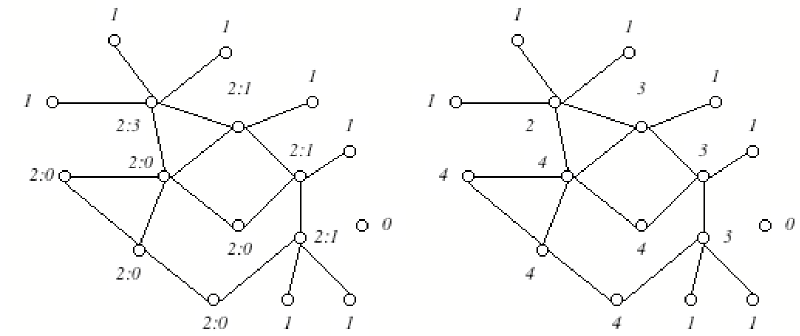
Consider the following example:



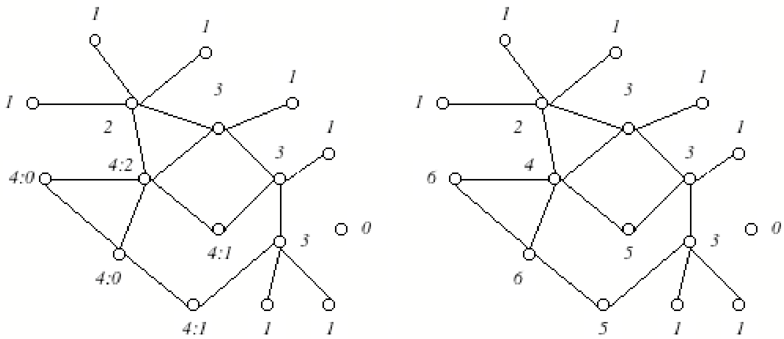
Start



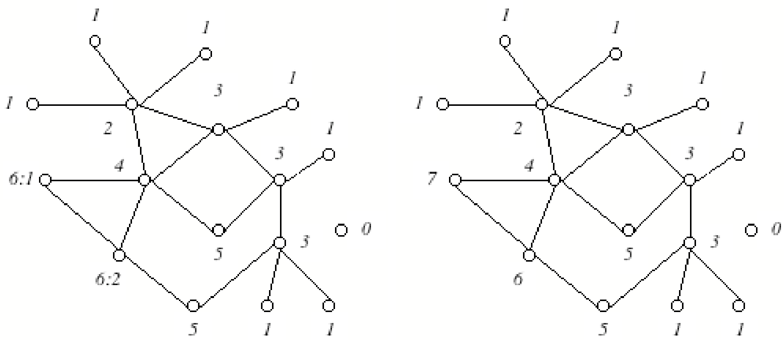
a)



b) (number of edges with vertices of lesser levels after double comma)



b) (number of edges with vertices of lesser levels after double comma)



b) (number of edges with vertices of lesser levels after double comma)

2. From Graphs to Standard Sentence Logic

Given any graph G, let every vertex be a standard sentence logic *formula*. Given the edge xy of G, if $\Lambda(x) > \Lambda(y)$ then let x be an *overformula* of y and y an *underformula* of x, if $\Lambda(y) > \Lambda(x)$ then vice versa, finally, if $\Lambda(x) = \Lambda(y)$ then let x and y be equivalent between them.

So we can assign to the vertices of any graph truth values and truth functions because they are logic formulas. Given any graph G and its vertex x, apply the following rules:

1) All the elements of $\Gamma(x)$ (elements of G that are adjacent to x) that have the same level of x have the same truth value of x.

2) If x has truth value *1* (true) then all the elements of $\Gamma(x)$ that have a level which is not equal to x have truth value *0* (false).

3) Consider the set which contains x and all its underformulas. If all its elements except one are false then the remaining element has to be true.

It is very easy to prove that the rules 1), 2), 3) can be reformulated in this way:

1a) Given a graph G, an edge between two vertices (i. e two sentences) with the same level is the connective '≡'.

2a) Any vertex is an atomic sentence when it has no underformulas.

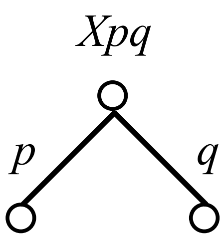
3a) Any vertex x which has the sole vertices $y_1, \ldots, y_n$ as underformulas corresponds to the sentence $\sim y_1 \wedge \ldots \wedge \sim y_n$.

3. Dyadic Connectives as Graphs

Standard sentence logic defines sixteen dyadic connectives:

Standard notation	Polish notation
$(p \vee \sim p) \wedge (q \vee \sim q)$	*Vpq*
$p \vee q$	*Apq*
$q \supset p$	*Bpq*
$p \supset q$	*Cpq*
$\sim p \vee \sim q$	*Dpq*
$(p \supset q) \wedge (q \supset p)$	*Epq*
$\sim p \wedge (q \vee \sim q)$	*Fpq*
$(p \vee \sim p) \wedge \sim q$	*Gpq*
$(p \vee \sim p) \wedge q$	*Hpq*
$p \wedge (q \vee \sim q)$	*Ipq*
$(p \wedge \sim q) \vee (q \wedge \sim p)$	*Jpq*
$p \wedge q$	*Kpq*
$p \wedge \sim q$	*Lpq*
$\sim p \wedge q$	*Mpq*
$\sim p \wedge \sim q$	*Xpq*
$(p \wedge \sim p) \vee (q \wedge \sim q)$	*Opq*

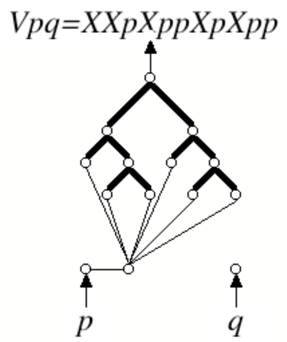
All these dyadic connectives are representable as graphs in various ways. In fact, it is very easy to prove that by 1a),2a),3a) we can deduce:

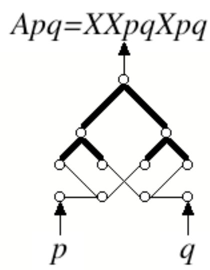


but *Xpq* is the dyadic connective *NOR*, i. e. $\sim p \wedge \sim q$. The dyadic connective *NOR* can represent every other connective. Thus:

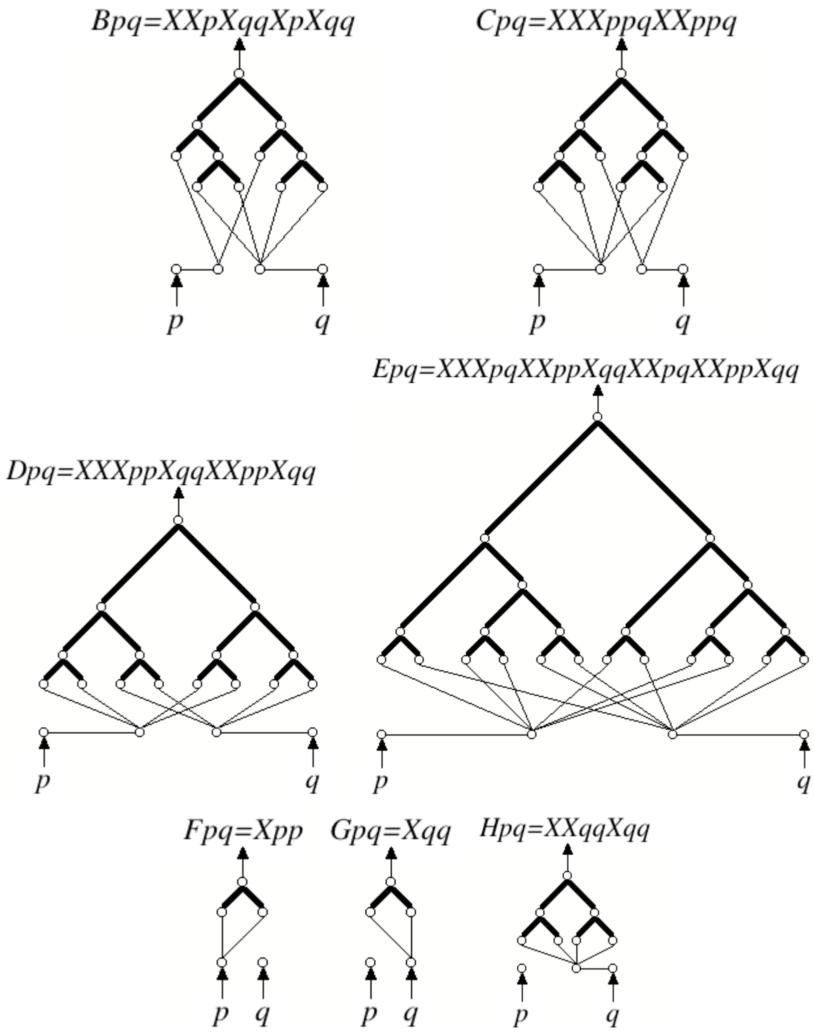
Dyadic connectives	Their definitions in *NOR* terms
Vpq	*XXpXppXpXpp*
Apq	*XXpqXpq*
Bpq	*XXpXqqXpXqq*
Cpq	*XXXppqXXppq*
Dpq	*XXXppXqqXXppXqq*
Epq	*XXXpqXXppXqqXXpqXXppXqq*
Fpq	*Xpp*
Gpq	*Xqq*
Hpq	*XXqqXqq*
Ipq	*XXppXpp*
Jpq	*XXpqXXppXqq*
Kpq	*XXppXqq*
Lpq	*XXppq*
Mpq	*XpXqq*
Xpq	*Xpq*
Opq	*XpXpp*

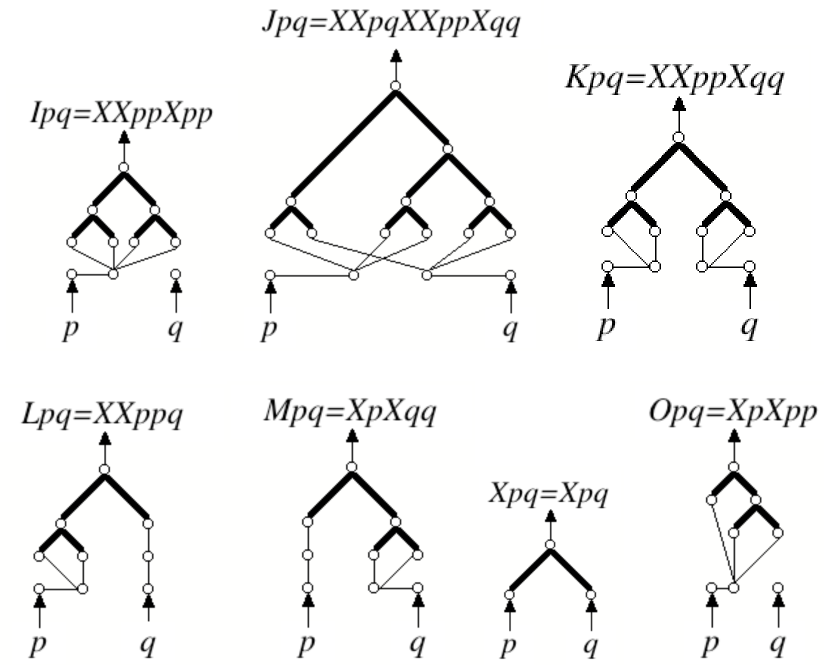
So we can build a graph representation of all the dyadic connectives. Among the possible representations we have:[1]





[1] To obtain a better comprehension some edges are bold and the dimensions of the graphs are different among them. Observe that these graphs are not oriented.


Bpq=XXpXqqXpXqq
p
q
Cpq=XXXppqXXppq
p
q
Epq=XXXpqXXppXqqXXpqXXppXqq
Dpq=XXXppXqqXXppXqq
p
q
p
q
Fpq=Xpp
p
q
Gpq=Xqq
p
q
Hpq=XXqqXqq
p
q



4. References

Bollobás B.
Graph Theory,
Springer-Verlag, New York, U.S.A., 1979

Malatesta M.
Logistica,
vol II: *Le tautologie L'interdefinibilità dei funtori*,
L.E.R., Napoli, Italy, 1978

Malatesta M.
The Primary Logic,
Gracewing Fowler Wright Books, Leominster, U.K.,1997