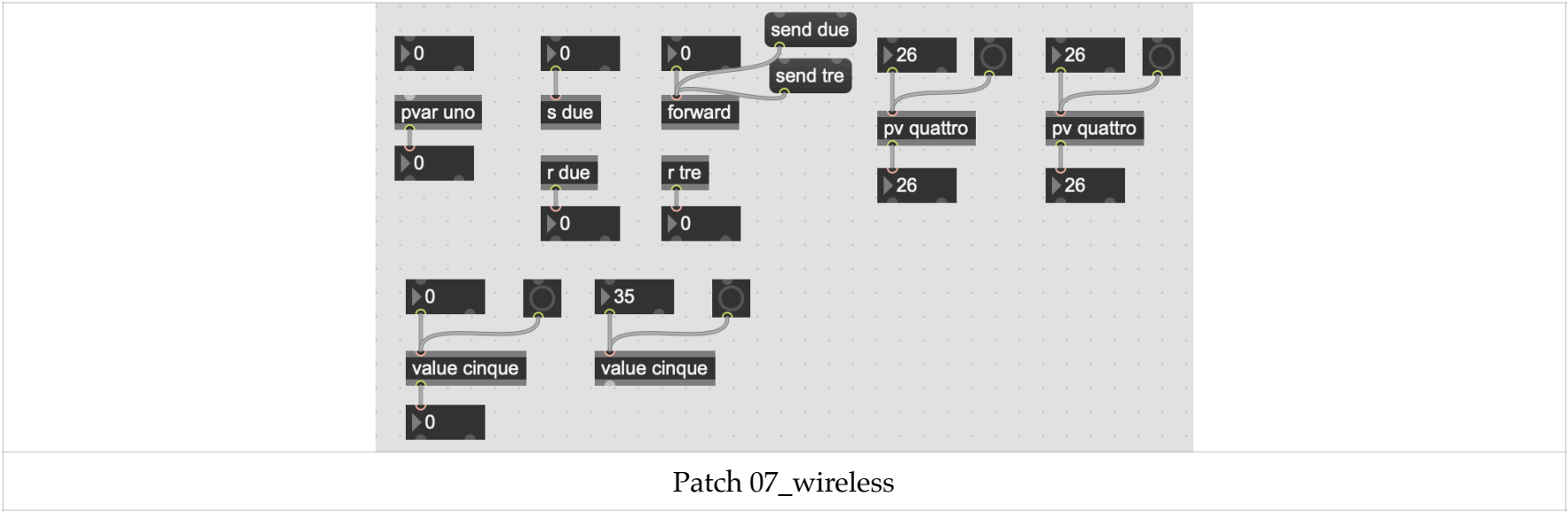


07_MESSAGGI WIRELESS



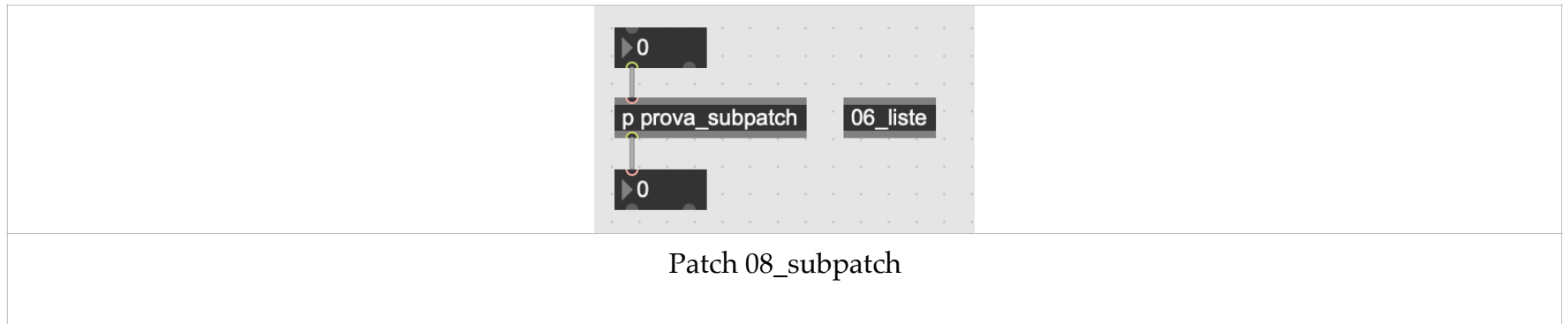
LISTA OGGETTI	LISTA MESSAGGI
<ol style="list-style-type: none"> 1. pvar 2. send 3. recive 4. forward 5. Pv 6. Value 	<ol style="list-style-type: none"> 1. bang 2. Number box 3. send

In Max è possibile spedire messaggi senza connettere i cavi. Gli oggetti che permettono questo sono “**send**” e “**recive**”, “**pvar**”, “**value**”, “**pv**” e “**forwad**”. Tutti questi oggetti sono simili, l’unica differenza sta nella possibilità di poter comunicare anche con altre patch, subpatch o solo nello stesso patcher. Gli oggetti “**send**” e “**recive**” sono gli oggetti più potenti che possono comunicare con altri “**send**” e “**recive**” con lo stesso argomento in qualsiasi patch o subpatch aperta.

L’oggetto “**forwad**” è un equivalente di “**recive**” ma con la possibilità di cambiare argomento (e quindi send di destinazione) inviando un messaggio con la keyword “**send**” ed il nome dei send. L’oggetto “**pvar**” è un equivalente di “**recive**”, e riceve messaggi da tutti quegli oggetti che hanno lo stesso *scripting name* del suo argomento (lo scripting name di un oggetto si cambia dall’*inspector*). Questo oggetto può comunicare solo all’interno dello stesso parcher, neanche tra le subpatch.

Gli oggetti “**pv**” e “**value**” sono equivalenti, ricevono ed inviano messaggi con altri oggetti “**pv**” e “**value**” con lo stesso argomento, memorizzando il valore e restituendolo dopo aver ricevuto un bang. La differenza è che Value funziona anche su patch diverse, mentre “**pv**” solo nella stessa patch (quindi all’interno del patcher e nelle subpatch).

08_SUBPATCH E ABSTRACTION



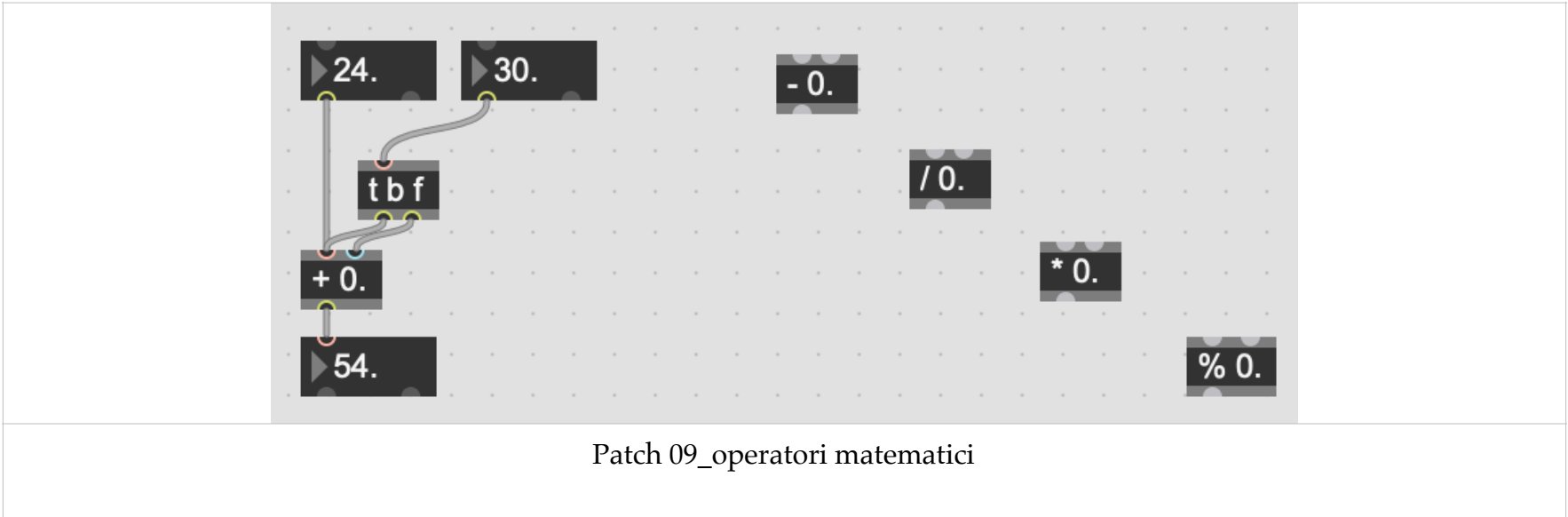
LISTA OGGETTI	LISTA MESSAGGI
1. patcher 2. abstraction	1. Number box

In Max è possibile creare delle patch contenute all'interno di un oggetto chiamato "patcher". Queste tipologie di patch si chiamano "subpatch". Quando creiamo una subpatch dobbiamo inserire al suo interno almeno due oggetti: inlet e outlet, in modo da poter creare degli ingressi e delle uscite per i messaggi che verranno elaborati dalla nostra patch. Ovviamente, selezionando una parte di una patch già scritta, tramite il medù edit - encapsulate, è possibile rinchiudere la parte della patch in un patcher in maniera automatica.

Se noi salviamo una patch all'interno di una cartella e iniziamo a lavorare su un'altra patch che salviamo all'interno della stessa cartella, possiamo richiamare come oggetto la patch salvata precedentemente. È possibile, tramite il menù option - file preference, chiedere a Max di cercare le patch in una cartella specifica, in questo modo sarà possibile richiamare le abstraction anche se non sono nella stessa cartella della nuova patch.

(nell'esempio ho richiamato l'oggetto 06_liste, se hai scaricato tutte le patch nella stessa cartella, cliccandoci sopra due volte puoi aprire la patch)

09_MATEMATICA IN MAX (1) - OPERATORI MATEMATICI

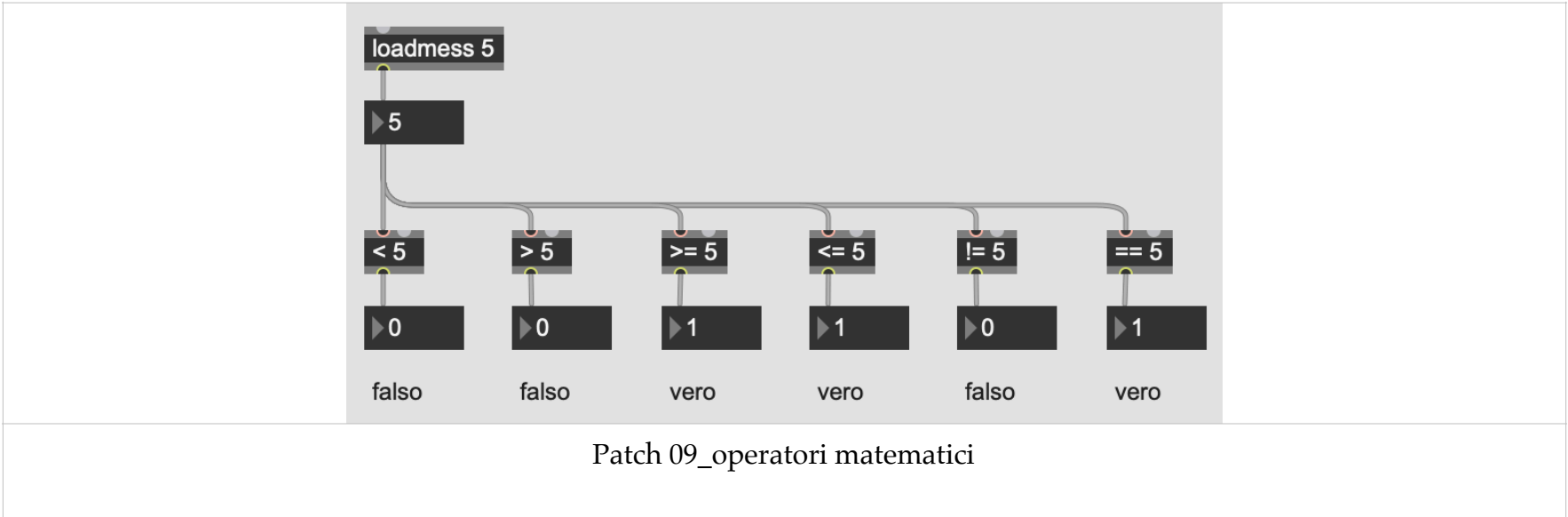


LISTA OGGETTI	LISTA MESSAGGI
1. +	1. Number box
2. -	
3. /	
4. *	
5. %	

Gli operatori matematici sono degli oggetti che eseguono le operazioni matematiche di base: addizione “+”, sottrazione “-“, moltiplicazione “*”, divisione “/“ e modulo “%”. Tutti gli operatori matematici, se non viene specificato nessun argomento, lavorano con numeri integer. Per lavorare con numeri float dobbiamo specificarlo inserendo come argomento un numero con la virgola come 0.

Gli operatori hanno due inlet, in quello di sinistra entrano i numeri da operare con l’argomento dell’oggetto, nell’inlet sinistro serve a cambiare l’argomento, tuttavia l’operazione avviene solo quando l’oggetto riceve un messaggio o un bang nell’inlet sinistro. Per eliminare il problema possiamo inserire un bang che invii un messaggio di bang all’inlet caldo successivamente aver cambiato l’argomento dall’inlet freddo.

09BIS_MATEMATICA IN MAX (1) - OPERATORI RELAZIONALI

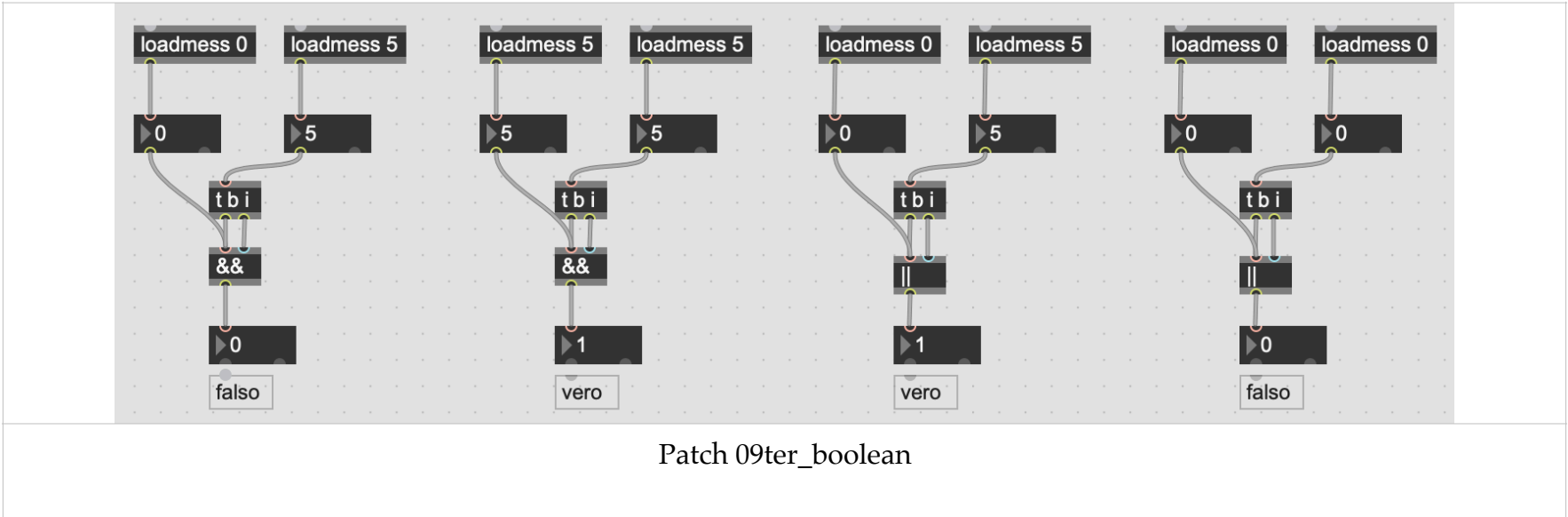


LISTA OGGETTI	LISTA MESSAGGI
<div>1. < (minore di...)</div> <div>2. > (maggiore di...)</div> <div>3. != (diverso da...)</div> <div>4. == (uguale a...)</div> <div>5. >= (maggiore uguale a...)</div> <div>6. <= (minore uguale a...)</div> <div>7. Loadmess</div>	<div>1. Number box</div>

Gli operatori relazioni confrontano due valori e stabiliscono se questi si trovino in una certa relazione. Per ogni relazione esiste un diverso operatore il cui risultato sarà vero (1) o falso (0).

L'oggetto “loadmess” serve a inviare il messaggio contenuto nell’argomento all’apertura della patch: in questo caso, appena si carica la patch, invierà il numero 5 al Number box collegato.

09TER_MATEMATICA IN MAX (1) - OPERATORI BOOLEANI



LISTA OGGETTI	LISTA MESSAGGI
1. && (and)	1. Number box
2. (or)	2. bang
3. Loadmess	
4. trigger	

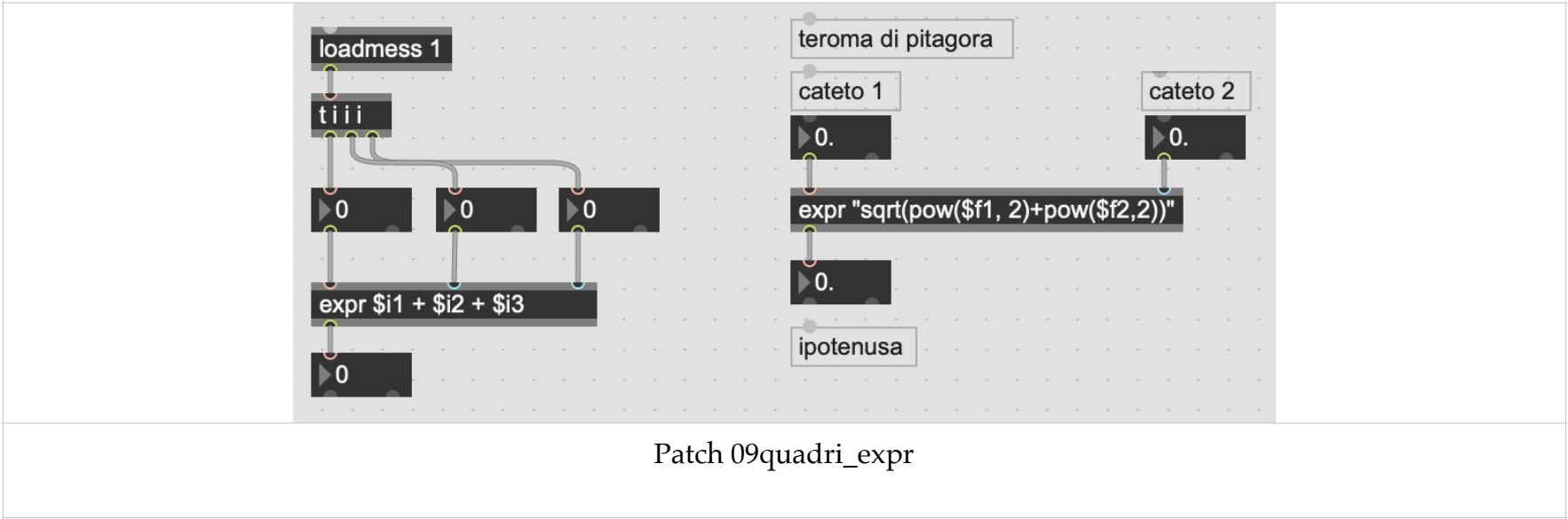
Gli operatori booleani servono a confrontare due affermazioni vere o false, e restituiscono un valore vero o falso a seconda del tipo di operatore.

In Max esistono solo gli operatori “&&” (and) e “||” (or):

- 1. And = restituisce vero se entrambi i valori in entrata sono veri, altrimenti falso.
- 2. Or = restituisce vero se almeno un valore in entrata è vero, altrimenti falso.

Restituire vero o falso vuol dire far uscire un 1 o uno 0, mentre un numero in entrata falso è 0 mentre vero un qualsiasi numero diverso da 0. Attenzione anche qui agli inlet caldi e freddi, è stato necessario infatti inserire un oggetto trigger.

09QUADRI_MATEMATICA IN MAX (1) - *EXPR*



LISTA OGGETTI	LISTA MESSAGGI
1. Expr 2. Trigger 3. Loadmess	1. Number box

“expr” è un oggetto che ci permette di costruire espressioni matematiche formate da una qualsiasi combinazioni di tutti gli operatori visti e le funzioni matematiche: l’argomento di questo oggetto è la funzione che vogliamo calcolare. Le variabili si segnano con il simbolo \$ seguite dal tipo di numero

L’oggetto “expr” ha solo il primo inlet caldo, questo vuol dire che calolerà la funzione solo se riceverà un “bang” dal primo inlet.