Informatics 1

Computation and Logic

Lecture 10 Refutation

Michael Fourman

When a system fails four Boolean *tests* RAGB may be performed to diagnose the error.
Different combinations of the results for these tests are the error states.
Each error code (1:8) applies to the set of error states satisfying the expression given in the table

1	¬R∧A∧¬G
2	¬R∧¬A∧¬G∧¬B
3	¬R∧¬G∧B
4	¬R∧G
5	R∧¬G
6	R∧G∧¬B
7	R∧¬A∧G
8	A∧B

Do these different error codes cover all possible error states?

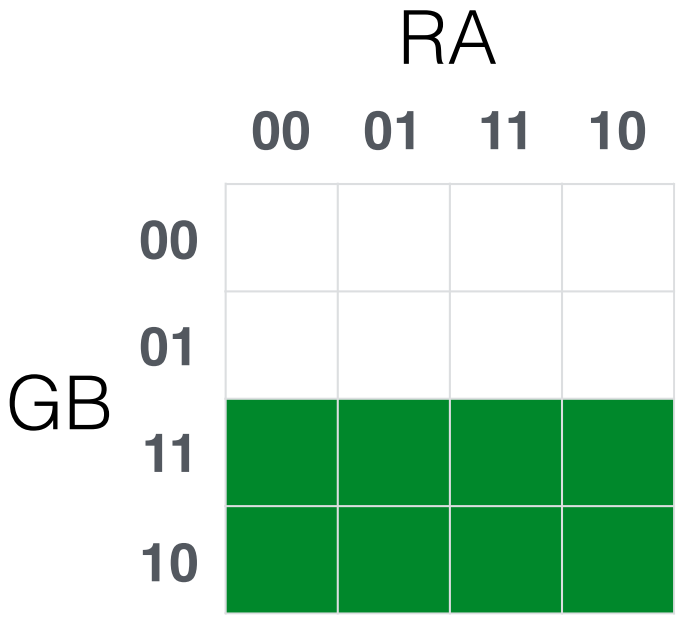
To determine the full error state takes time.

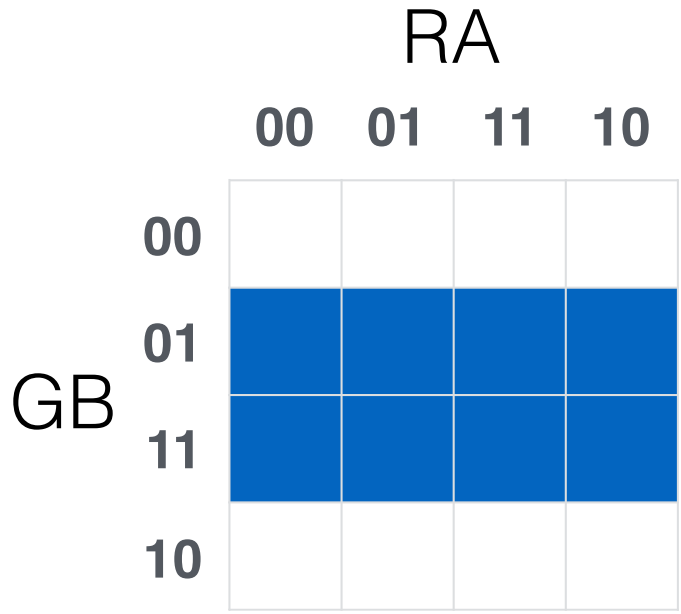
We can diagnose some error codes on the basis of only two or three tests, to save time.

Can you give an efficient procedure for error diagnosis?


RA
00
01
11
10
GB
00
01
11
10


RA
00
01
11
10
GB
00
01
11
10


RA
00
01
11
10
GB
00
01
11
10


RA
00
01
11
10
GB
00
01
11
10

1 ¬R∧A∧¬G

2 ¬R∧¬A∧¬G∧¬B

3 ¬R∧¬G∧B

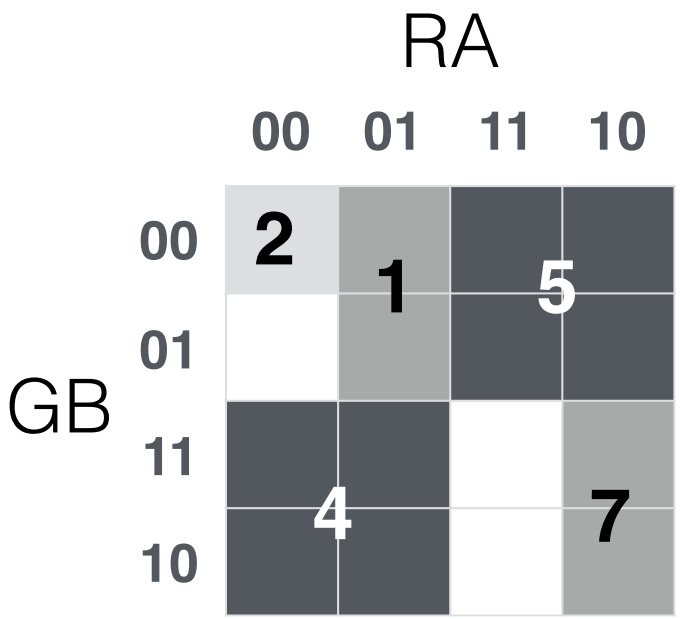
4 ¬R∧G

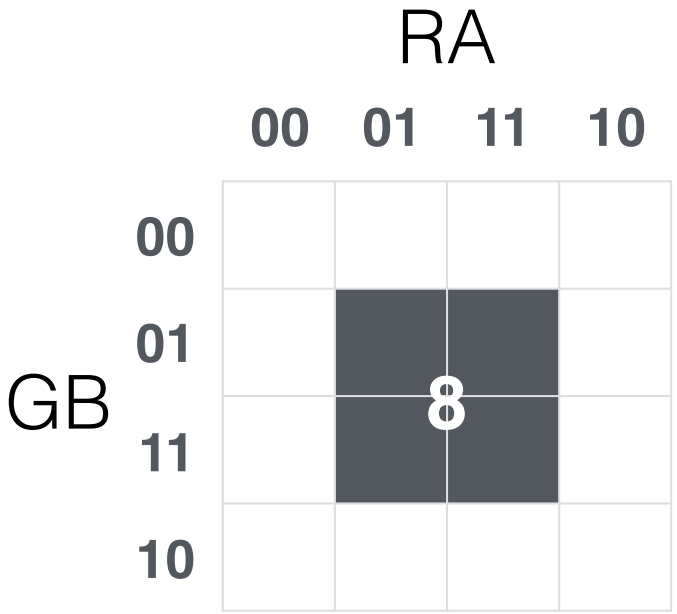
5 R∧¬G

6 R∧G∧¬B

7 R∧¬A∧G

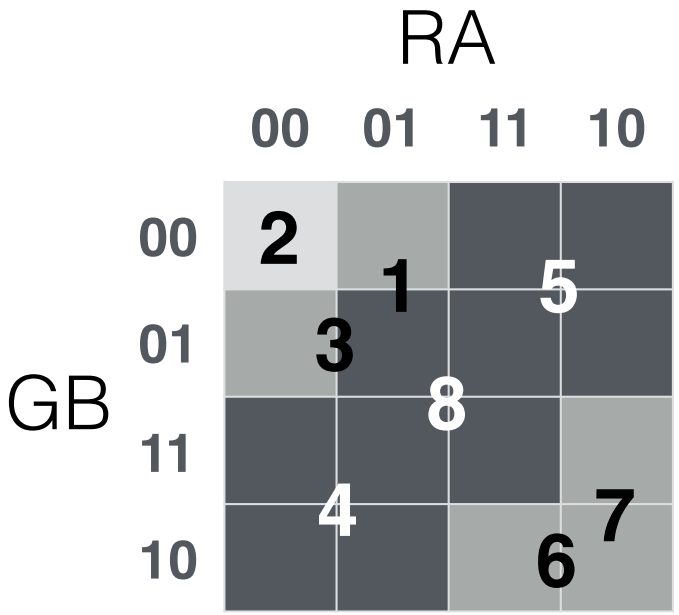
8 A∧B





The diagrams show that the different types of error cover all possible combinations of the four flags.
The following eight constraints describe errors not included in any type

¬(¬R∧A∧¬G)
¬(¬R∧¬A∧¬G∧¬B)
¬(¬R∧¬G∧B)
¬(¬R∧G)
¬(R∧¬G)
¬(R∧G∧¬B)
¬(R∧¬A∧G)
¬(A∧B)



These constraints are inconsistent since the eight types cover all possibilities.

To answer the first question we use the following eight constraints to describe errors not included in any type

$\neg(\neg R \wedge A \wedge \neg G)$
$\neg(\neg R \wedge \neg A \wedge \neg G \wedge \neg B)$
$\neg(\neg R \wedge \neg G \wedge B)$
$\neg(\neg R \wedge G)$
$\neg(R \wedge \neg G)$
$\neg(R \wedge G \wedge \neg B)$
$\neg(R \wedge \neg A \wedge G)$
$\neg(A \wedge B)$

If these constraints are inconsistent the eight types cover all possibilities.

To answer the first question we use the following eight constraints to describe errors not included in any type

$\neg(\neg R \wedge A \wedge \neg G)$
$\neg(\neg R \wedge \neg A \wedge \neg G \wedge \neg B)$
$\neg(\neg R \wedge \neg G \wedge B)$
$\neg(\neg R \wedge G)$
$\neg(R \wedge \neg G)$
$\neg(R \wedge G \wedge \neg B)$
$\neg(R \wedge \neg A \wedge G)$
$\neg(A \wedge B)$

We convert each constraint to a disjunctive clause, and use resolution to check whether any cases have been missed.

$R \vee \neg A \vee G$

$R \vee A \vee G \vee B$

$R \vee G \vee \neg B$

$R \vee \neg G$

$\neg R \vee G$

$\neg R \vee \neg G \vee B$

$\neg R \vee A \vee \neg G$

$\neg A \vee \neg B$

If these constraints are inconsistent the eight types cover all possibilities.

$$R \vee \neg A \vee G$$

$$R \vee A \vee G \vee B$$

$$R \vee G \vee \neg B$$

$$R \vee \neg G$$

$$\neg R \vee G$$

$$\neg R \vee \neg G \vee B$$

$$\neg R \vee A \vee \neg G$$

$$\neg A \vee \neg B$$

A

R∨¬A∨G

R∨A∨G∨B

R∨G∨¬B

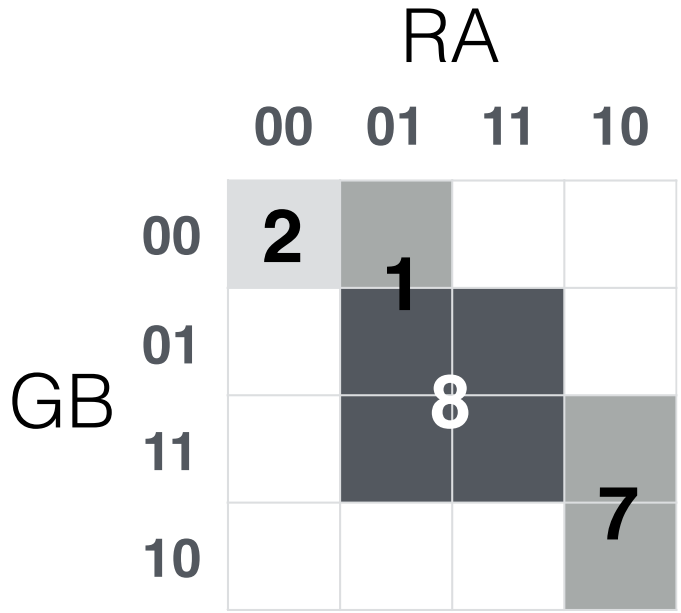
R∨¬G

¬R∨G

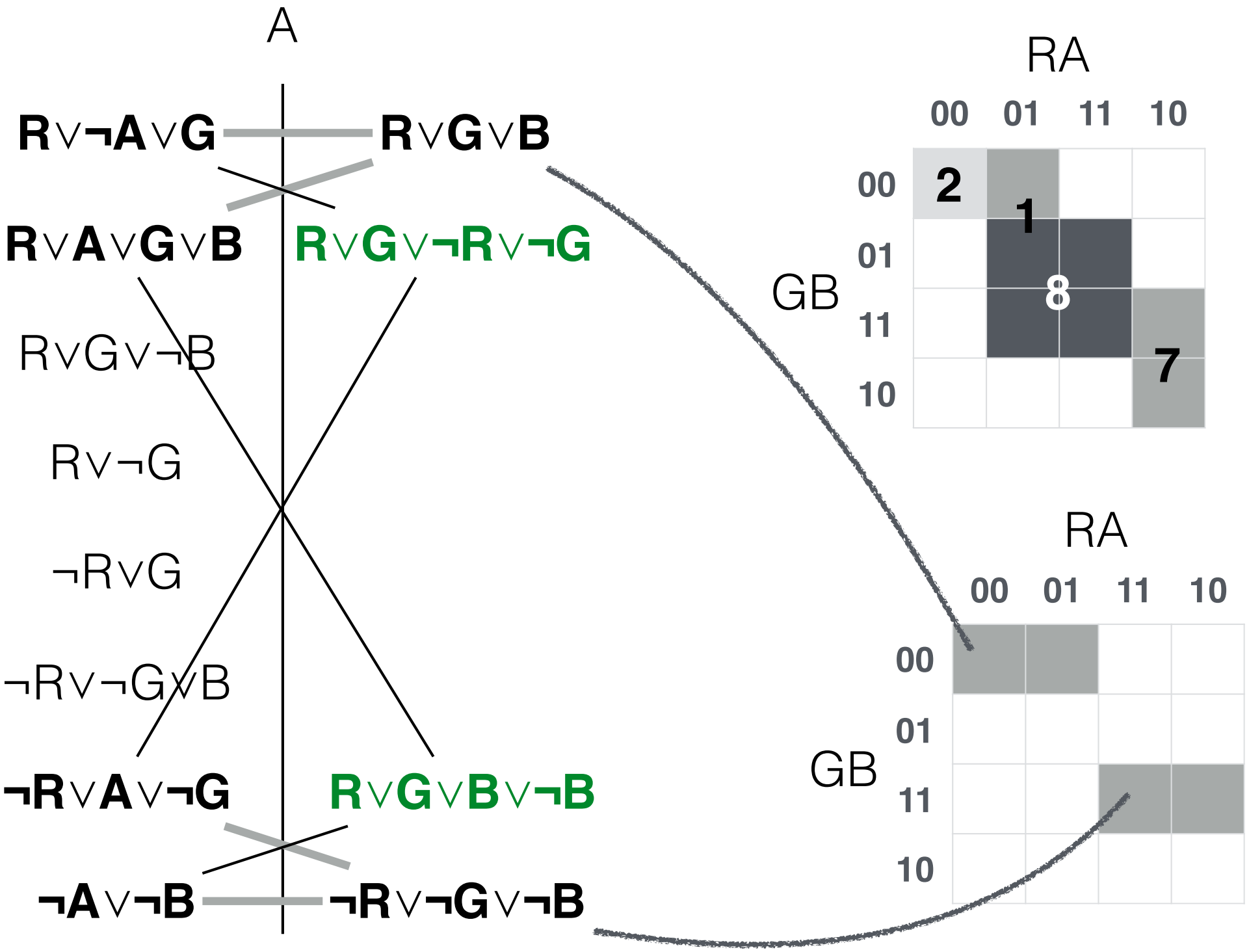
¬R∨¬G∨B

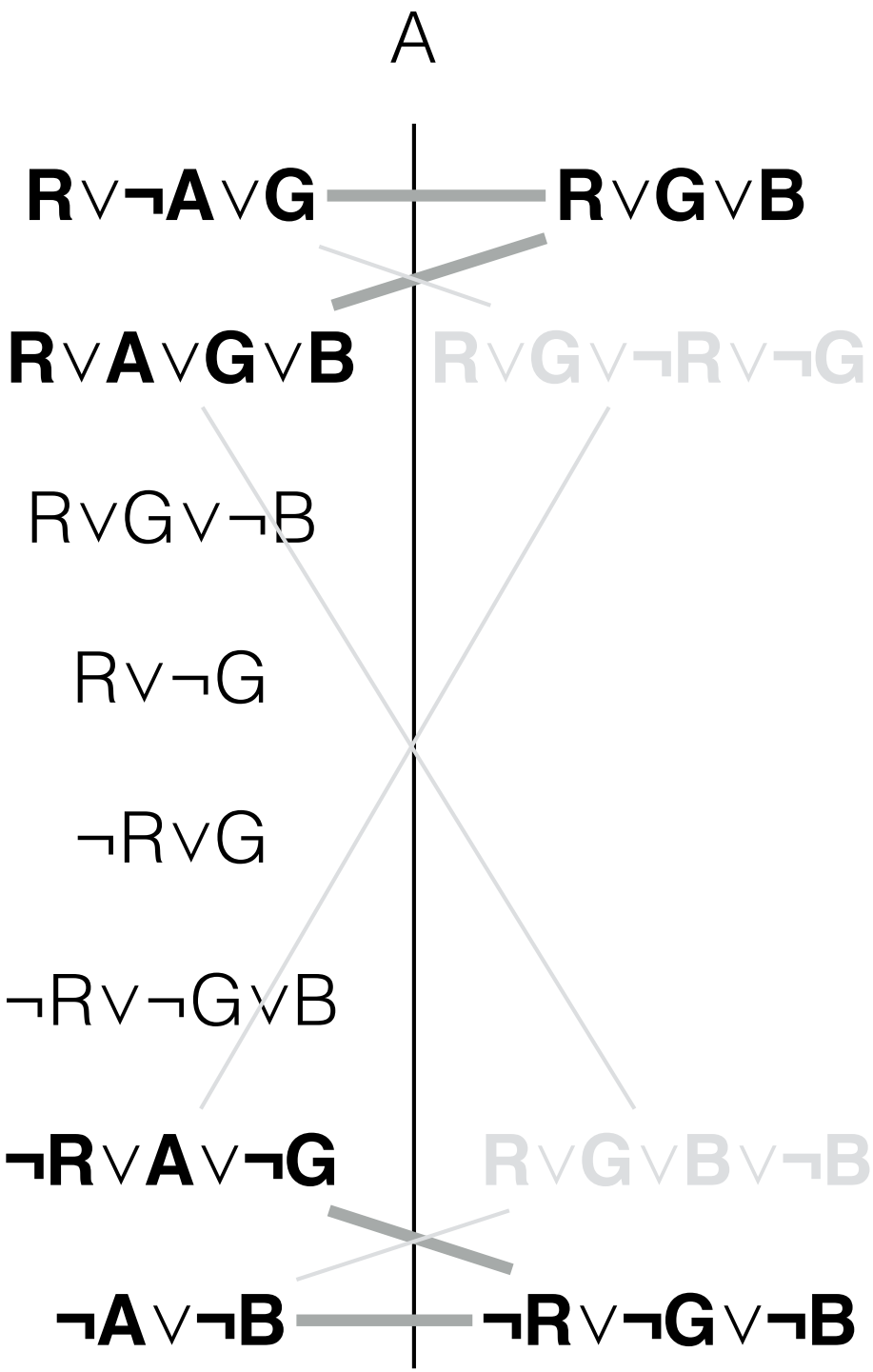
¬R∨A∨¬G

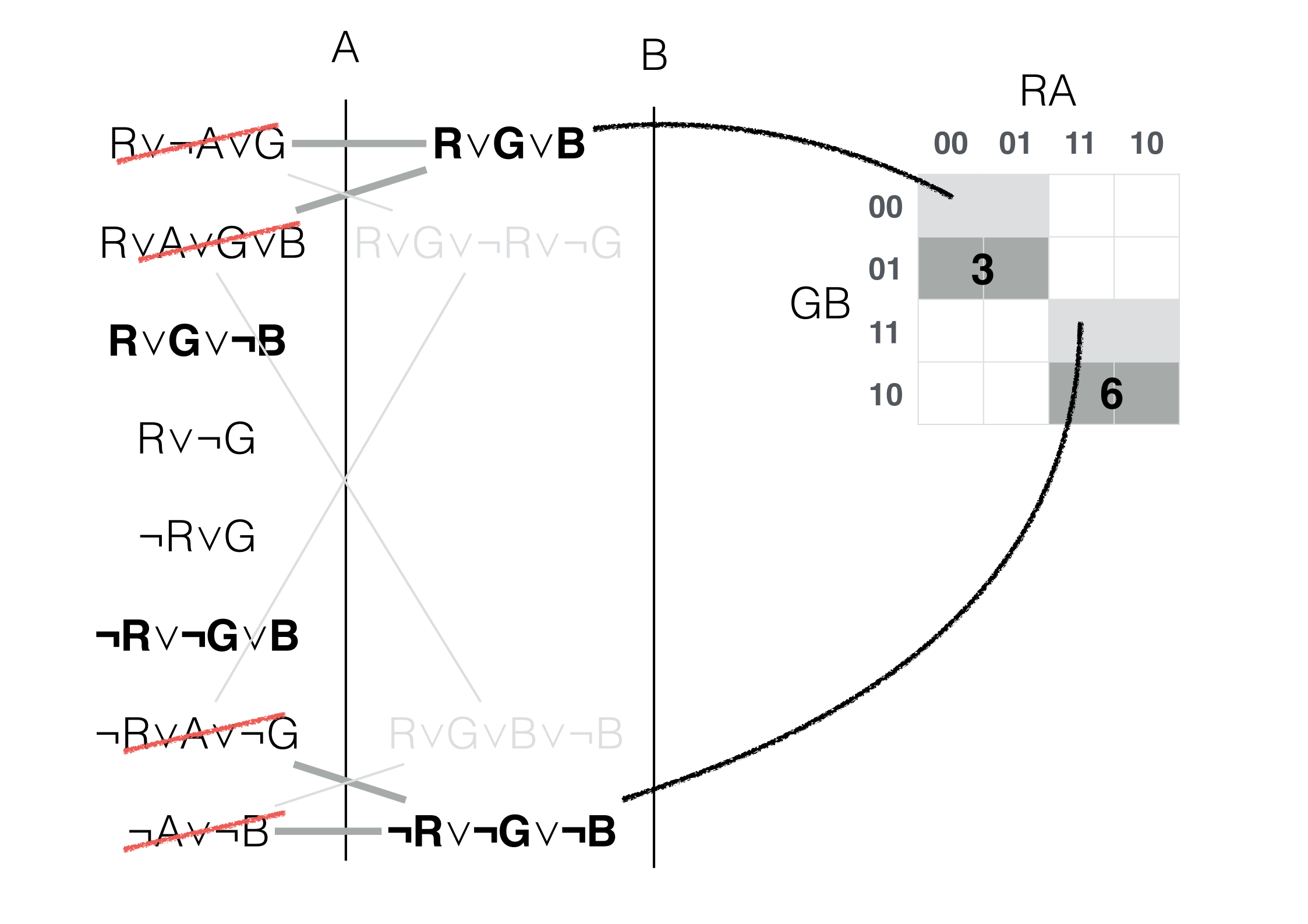
¬A∨¬B

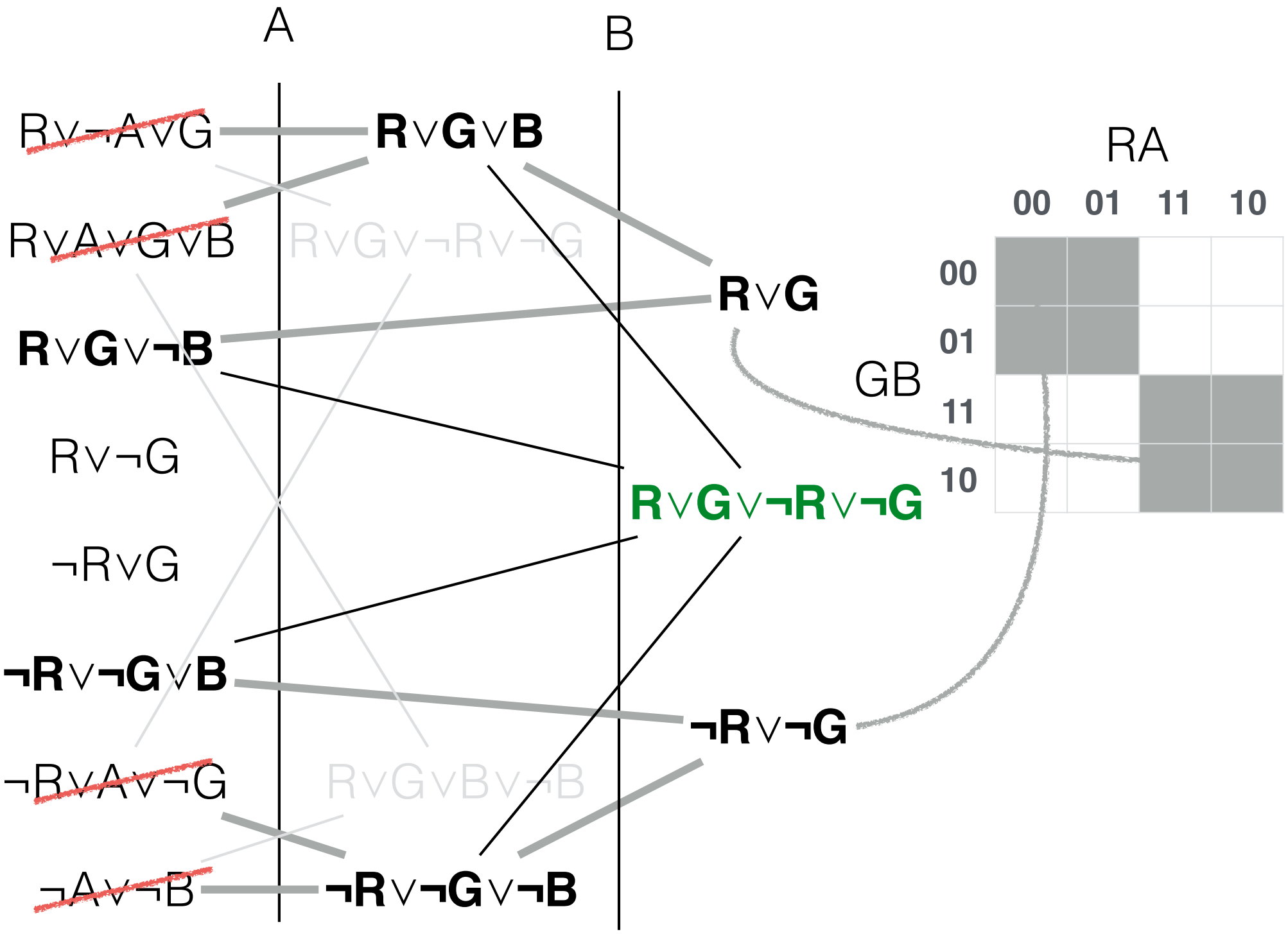


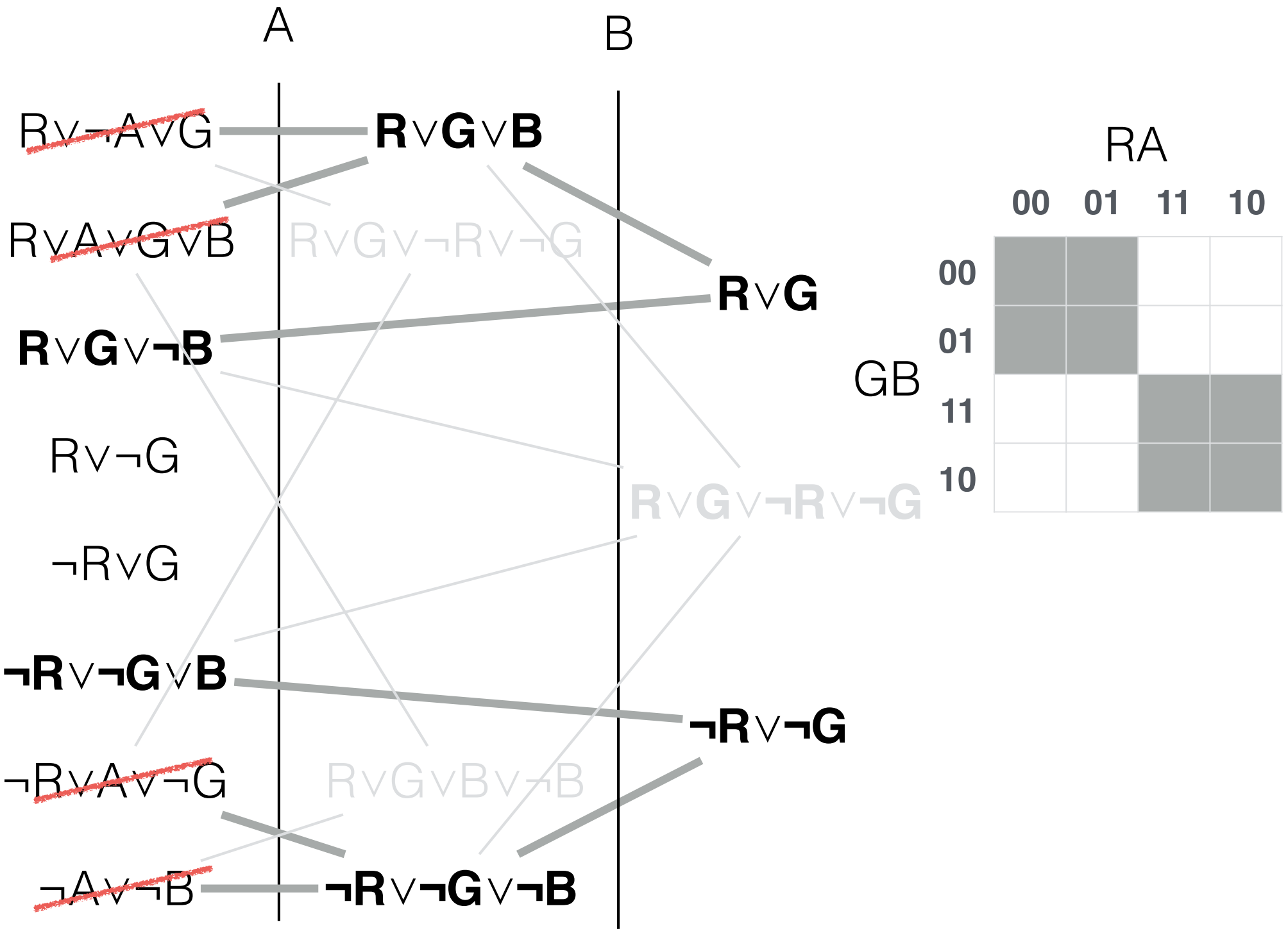
We use Karnaugh maps to show the states
excluded
by various clauses

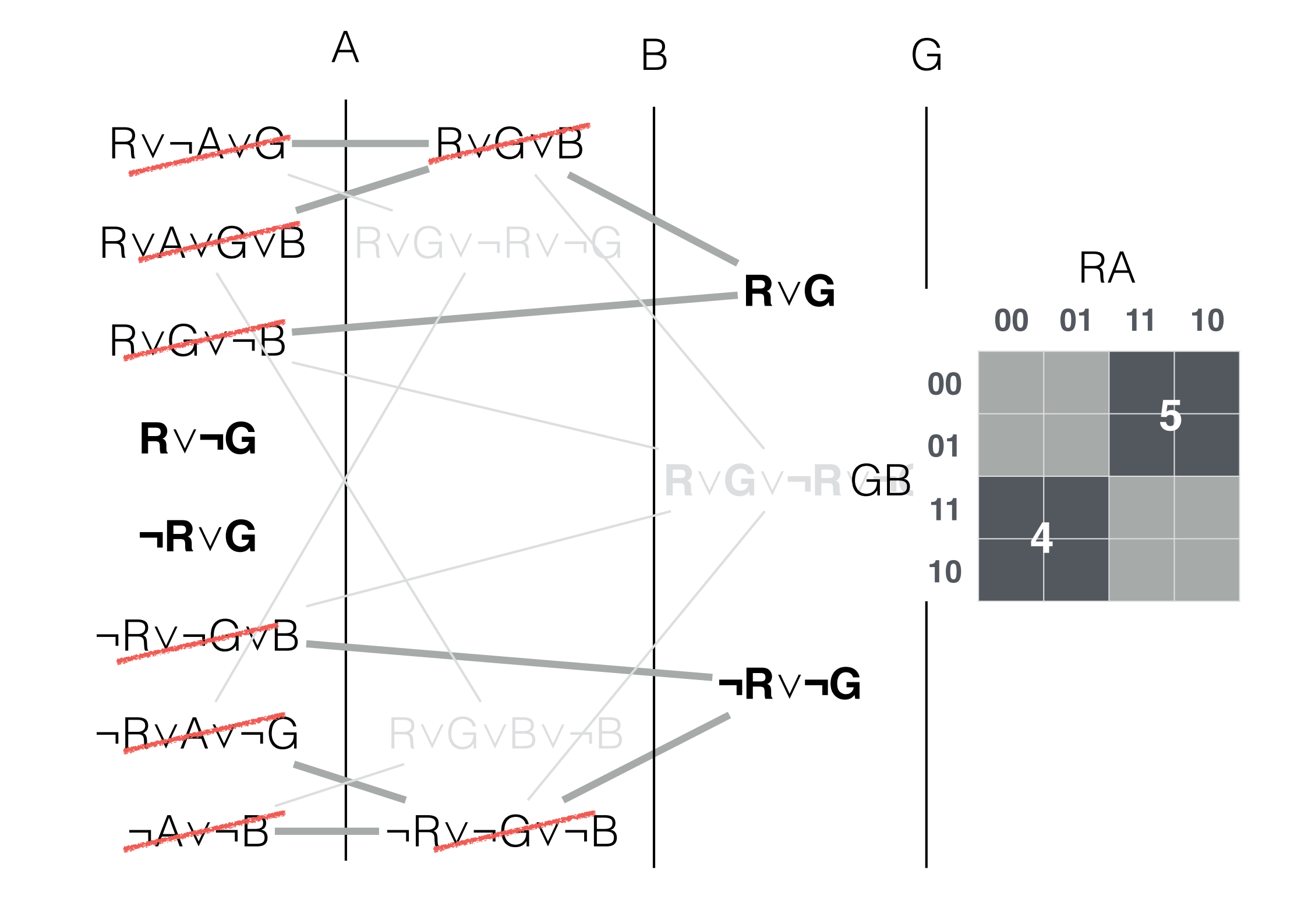

A
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G∨¬B
R∨¬G
¬R∨G
¬R∨¬G∨B
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
RA
00
01
11
10
GB
2
1
8
7
RA
00
01
11
10
GB

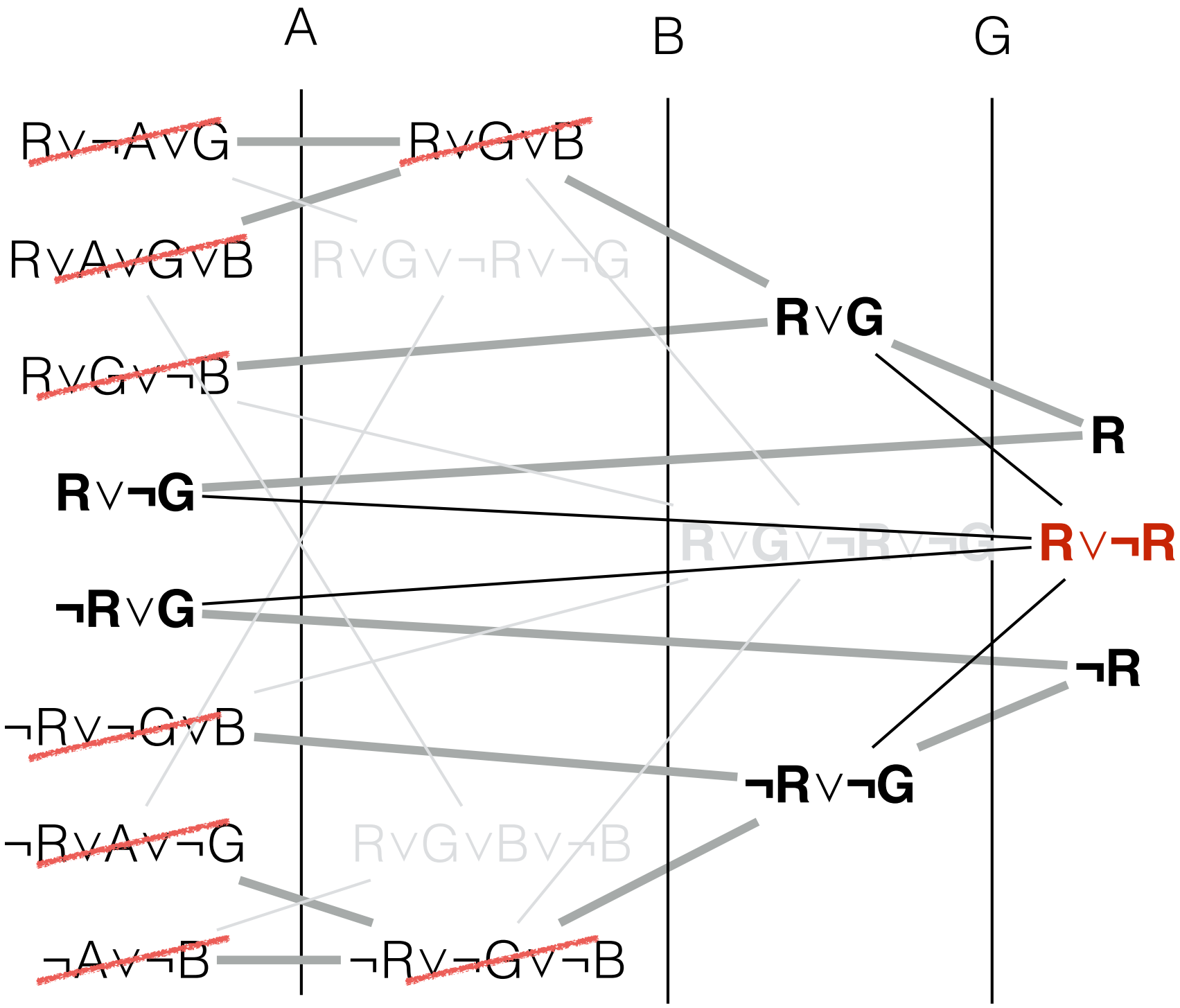

A
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G∨¬B
R∨¬G
¬R∨G
¬R∨¬G∨B
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B

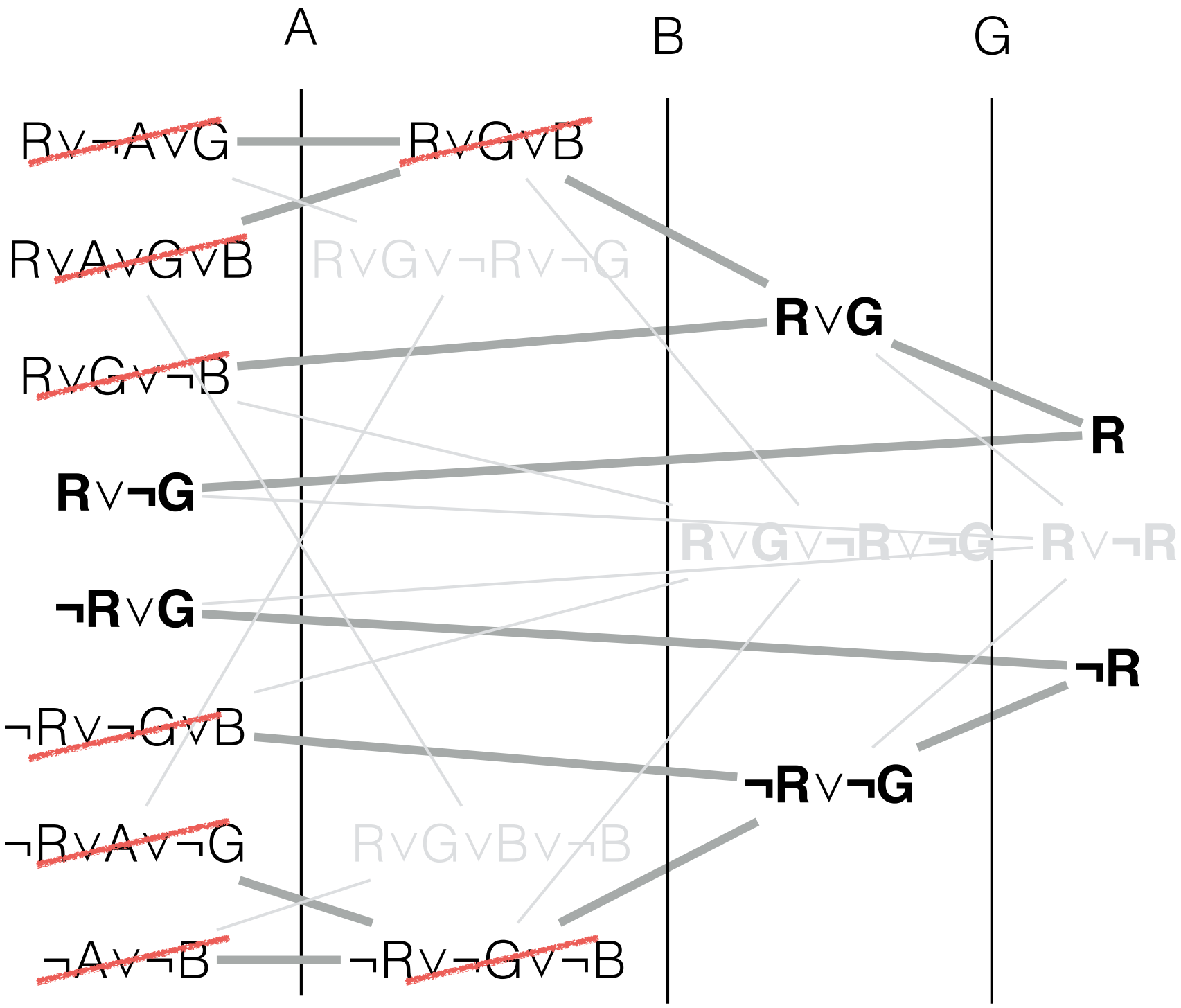

A
B
RA
00 01 11 10
GB
00
01
11
10
3
6
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G∨¬B
R∨¬G
¬R∨G
¬R∨¬G∨B
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B

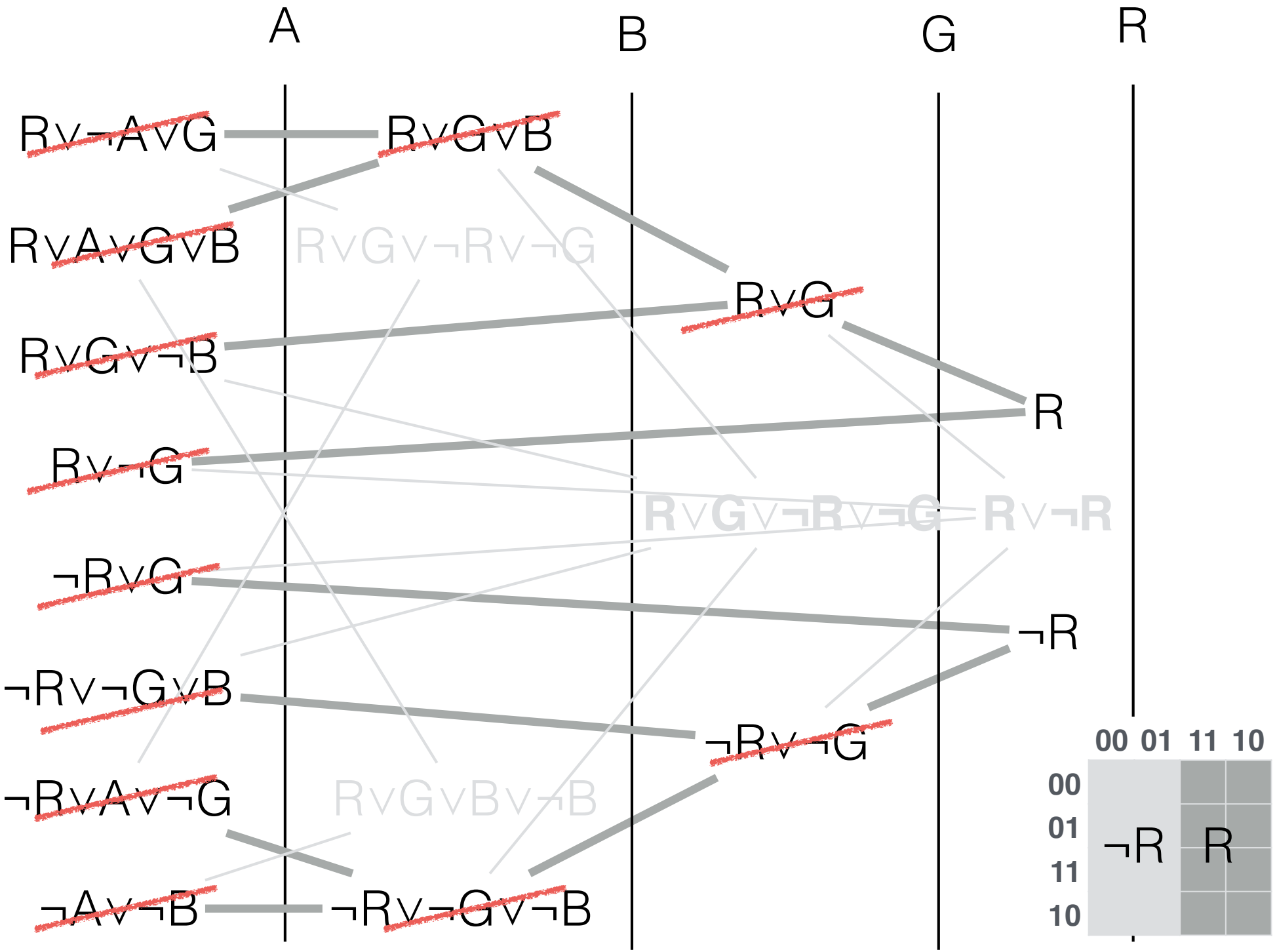

A
B
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R∨¬G
¬R∨G
R∨G∨¬R∨¬G
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
RA
00
01
11
10
GB
00
01
11
10

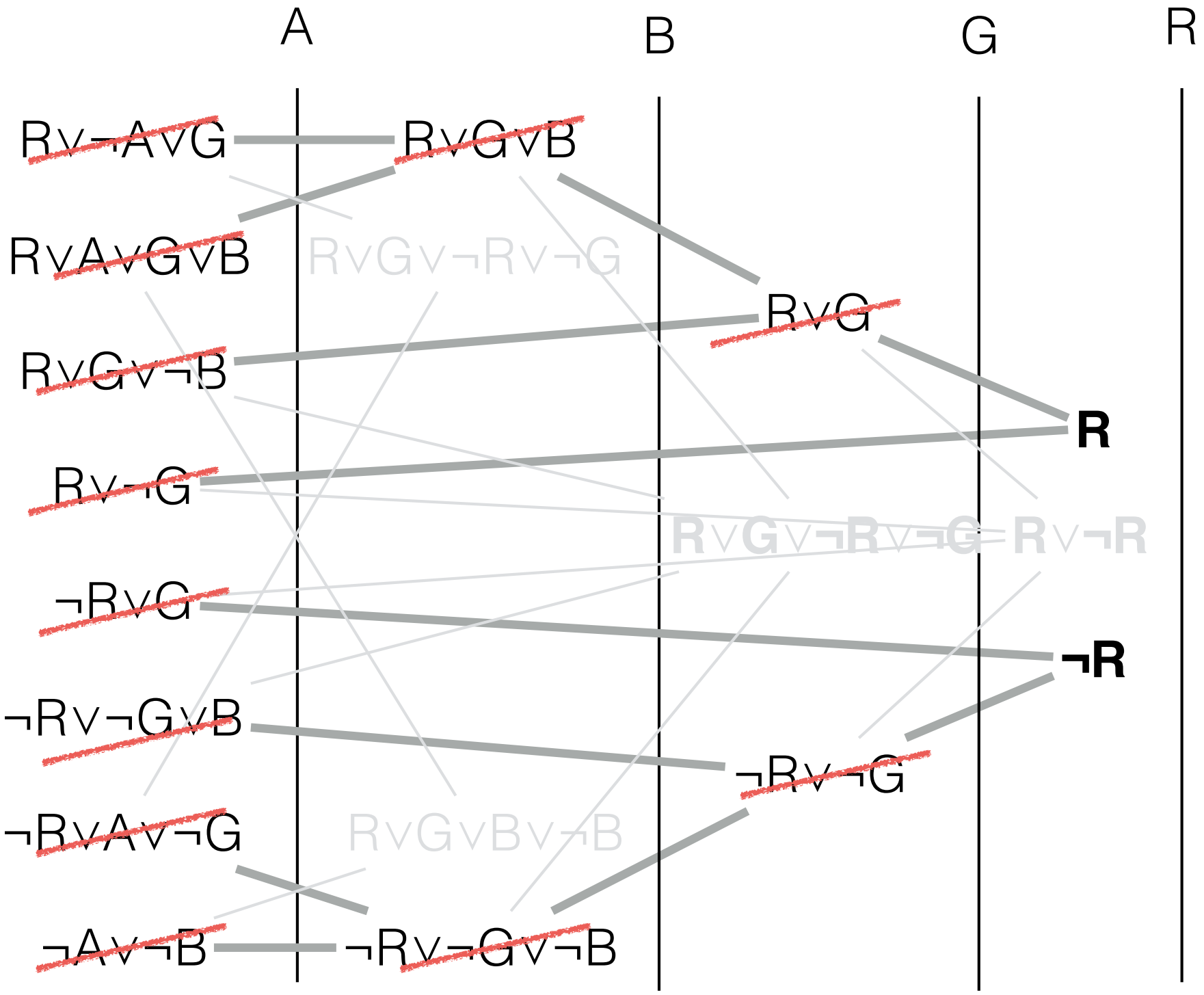

A
B
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R∨¬G
R∨G∨¬R∨¬G
¬R∨G
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
RA
00
01
11
10
GB
00
01
11
10

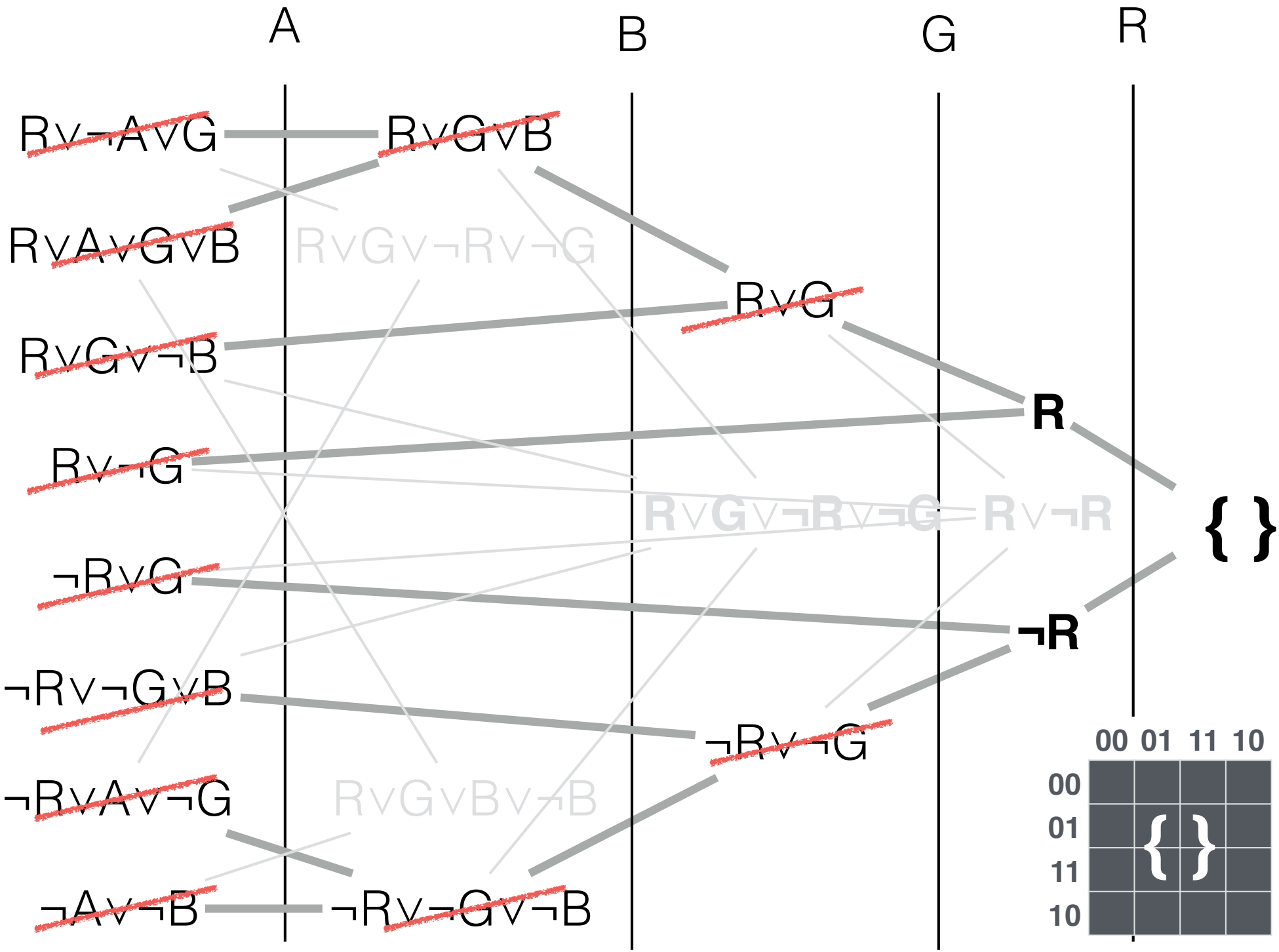

A
B
G
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R∨¬G
¬R∨G
R∨G∨¬R∨G
GB
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
RA
00
01
11
10
5
4

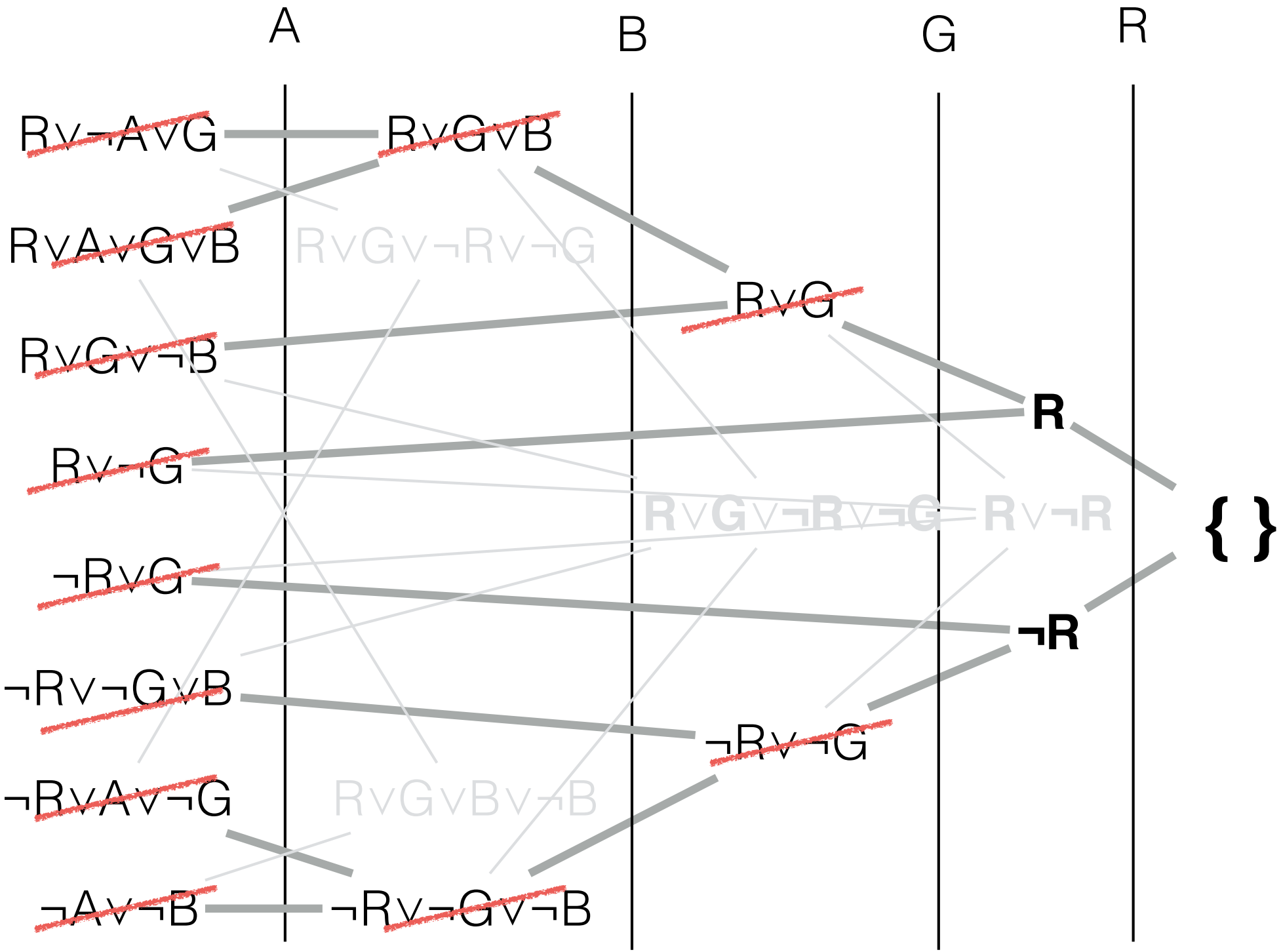

A
B
G
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B

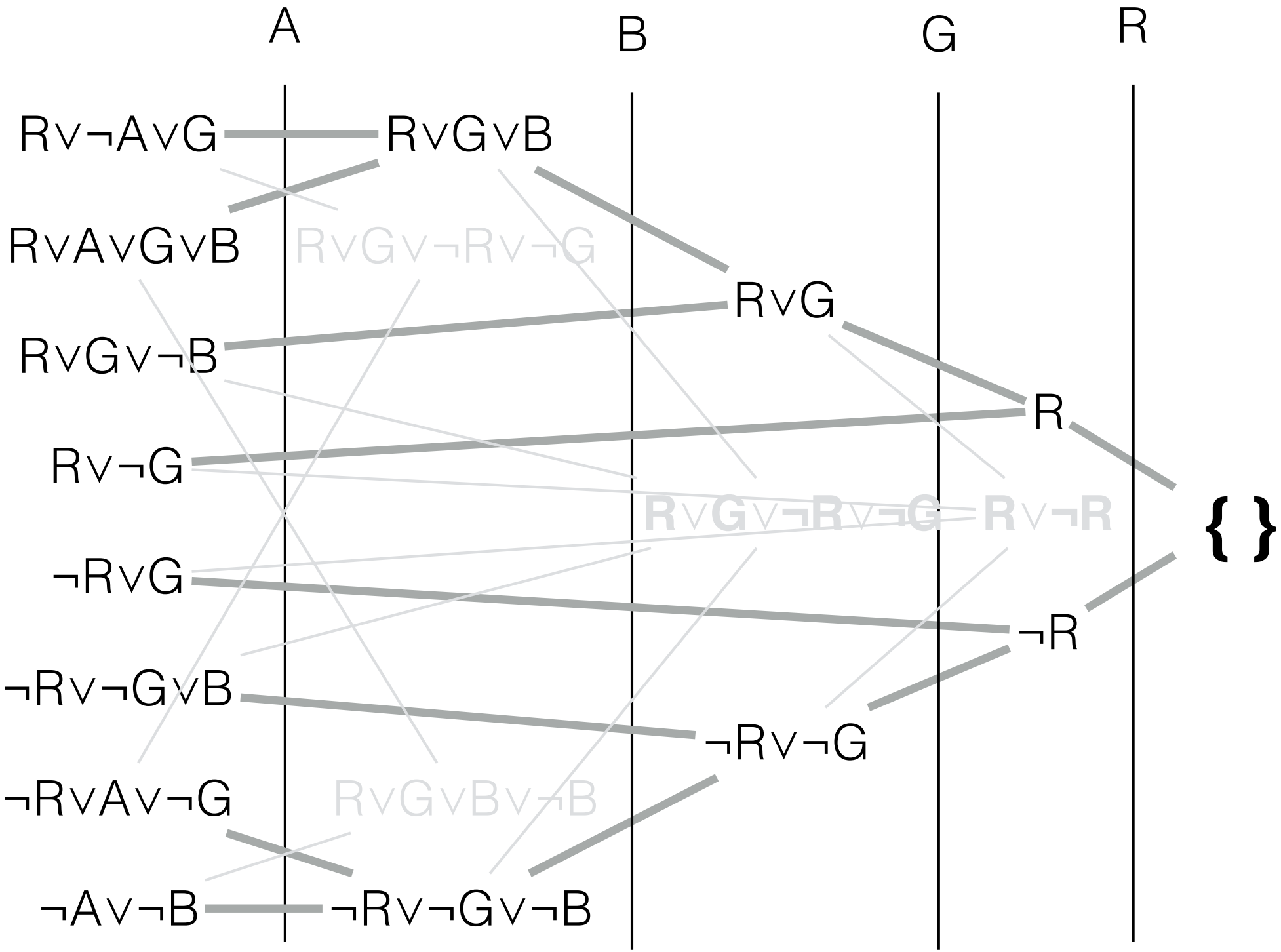

A
B
G
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B

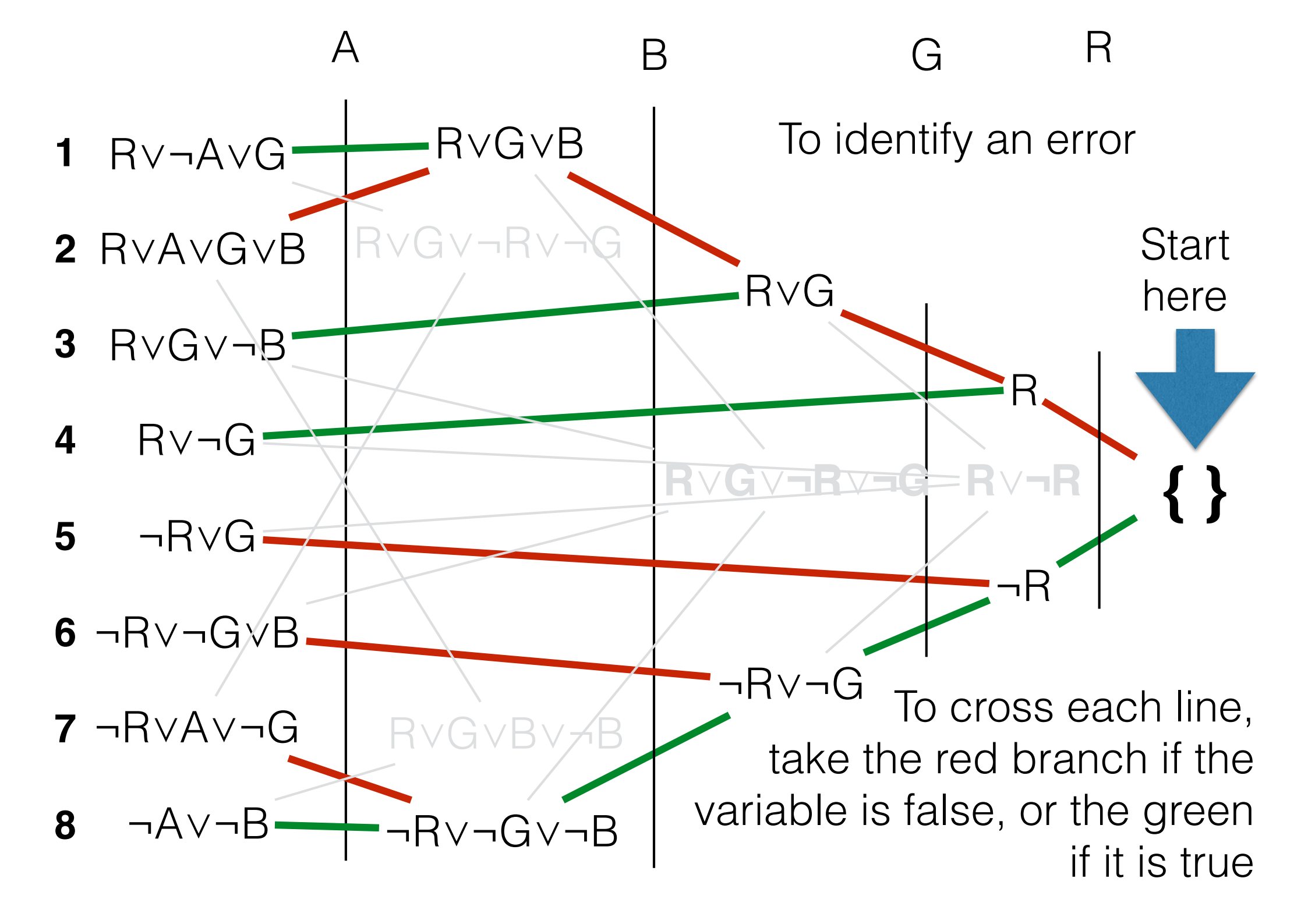

A
B
G
R
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
00 01 11 10
00
01
11
10
¬R
R

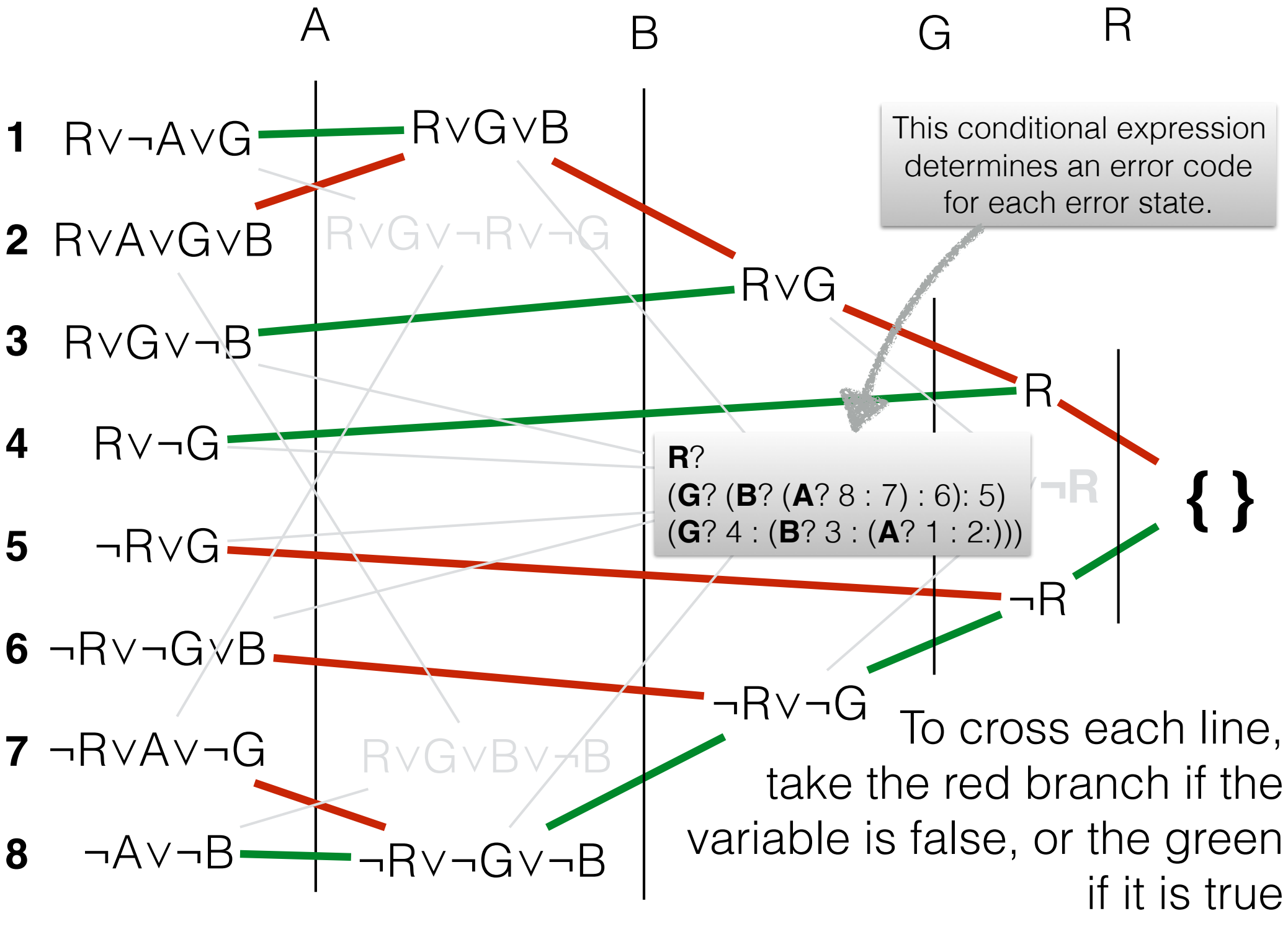

A
B
G
R
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B


A
B
G
R
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
{ }
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B
00 01 11 10
00
01
11
10
{ }


A
B
G
R
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
{ }
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B


A
B
G
R
R∨¬A∨G
R∨G∨B
R∨A∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬B
R
R∨¬G
R∨G∨¬R∨¬G
R∨¬R
{ }
¬R∨G
¬R
¬R∨¬G∨B
¬R∨¬G
¬R∨A∨¬G
R∨G∨B∨¬B
¬A∨¬B
¬R∨¬G∨¬B


A
B
G
R
1 R∨¬A∨G
2 R∨A∨G∨B
3 R∨G∨¬B
4 R∨¬G
5 ¬R∨G
6 ¬R∨¬G∨B
7 ¬R∨A∨¬G
8 ¬A∨¬B
R∨G∨B
R∨G∨¬R∨¬G
R∨G
R∨G∨¬R∨¬G
R∨¬R
R
¬R
¬R∨¬G
R∨G∨B∨¬B
¬R∨¬G∨¬B
To identify an error
Start here
{ }
To cross each line, take the red branch if the variable is false, or the green if it is true


A
B
G
R
1 R∨¬A∨G
2 R∨A∨G∨B
3 R∨G∨¬B
4 R∨¬G
5 ¬R∨G
6 ¬R∨¬G∨B
7 ¬R∨A∨¬G
8 ¬A∨¬B
R∨G∨B
R∨G∨¬R∨¬G
R∨G
R
¬R
¬R∨¬G
R∨G∨B∨¬B
¬R∨¬G∨¬B
This conditional expression determines an error code for each error state.
R?
(G? (B? (A? 8 : 7) : 6): 5)
(G? 4 : (B? 3 : (A? 1 : 2:)))
{ }
To cross each line, take the red branch if the variable is false, or the green if it is true