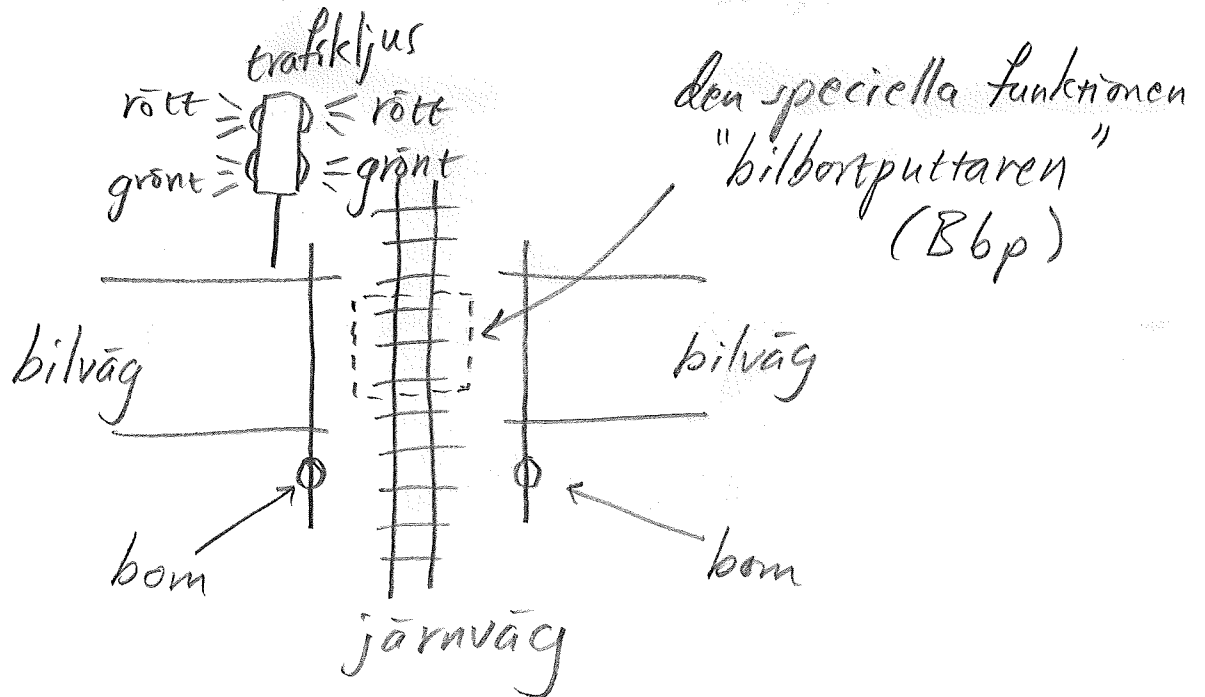


Illustration av logiktillämpning

Figur som illustrerar situationen



Preliminära systemspecifikationer

1. Tåg kommer $\rightarrow$ Röd signal
2. $(\text{Röd signal} \wedge \text{ingen bil på spåret}) \rightarrow$ Bomar ner
3. $(\text{Röd signal} \wedge \text{bil på spåret}) \rightarrow$
Bilbortputtaren igång
4. Inget tåg kommer $\rightarrow$ Bomar upp
5. Bomar upp $\rightarrow$ Grön signal

Formell översättning av specifikationerna

1. $Tåg \rightarrow Röd$ (signal)
2. $(Röd \wedge \neg Bil) \rightarrow Bomar$ (ner)
3. $(Röd \wedge Bil) \rightarrow Bbp$ (bilbortputtaren)
4. $\neg Tåg \rightarrow \neg Bomar$ (dvs. Bomar upp)
5. $\neg Bomar \rightarrow Grön$

Åtminstone följande krav måste uppfyllas:

6. Grön signal $\rightarrow$ inget tåg kommer och bomarna är uppfällda.

Formell översättning av kravet:

6. $Grön \rightarrow (\neg Tåg \wedge \neg Bomar)$

Vi önskar nu bevisa att kravet är uppfyllt om specifikationerna är uppfyllda, dvs. vi önskar bevisa att 6 är en tautologisk konsekvens av $1 \wedge 2 \wedge 3 \wedge 4 \wedge 5$.

(3)

Detta gäller om och endast om satsen

$$S: (1 \wedge 2 \wedge 3 \wedge 4 \wedge 5) \rightarrow 6$$

är en tautologi, och detta gäller i sin tur om och endast om satsen

$$\neg S$$

inte är satisfierbar.

Vi skall se om det senare gäller eller ej.

Först omvandlar vi $\neg S$ till konjunktiv normalform (KNF). Motivationen till detta är att program som testar satisfierbarhet ofta gör om input till KNF.

$$\neg S \stackrel{\text{taut}}{\iff} \neg((1 \wedge 2 \wedge 3 \wedge 4 \wedge 5) \rightarrow 6)$$

$$\stackrel{\text{taut}}{\iff} 1 \wedge 2 \wedge 3 \wedge 4 \wedge 5 \wedge \neg 6 \stackrel{\text{taut}}{\iff}$$

(4)

$$\begin{aligned}
& (Tåg \rightarrow Röd) \wedge \\
& ((Röd \wedge \neg Bil) \rightarrow Bomar) \wedge \\
& ((Röd \wedge Bil) \rightarrow Bbp) \wedge \\
& (\neg Tåg \rightarrow \neg Bomar) \wedge \\
& (\neg Bomar \rightarrow Grön) \wedge \\
& \neg (Grön \rightarrow (\neg Tåg \wedge \neg Bomar)) \\
& \text{taut} \\
& \iff
\end{aligned}$$

$$\begin{aligned}
& (\neg Tåg \vee Röd) \wedge \\
& (\neg (Röd \wedge \neg Bil) \vee Bomar) \wedge \\
& (\neg (Röd \wedge Bil) \vee Bbp) \wedge \\
& (\neg \neg Tåg \vee \neg Bomar) \wedge \\
& (\neg \neg Bomar \vee Grön) \wedge \\
& (Grön \wedge \neg (\neg Tåg \wedge \neg Bomar)) \\
& \text{taut} \\
& \iff
\end{aligned}$$

5

$$(\neg \text{Tåg} \vee \text{Röd}) \wedge$$

$$((\neg \text{Röd} \vee \neg \text{Bil}) \vee \text{Bomar}) \wedge$$

$$((\neg \text{Röd} \vee \neg \text{Bil}) \vee \text{Bbp}) \wedge$$

$$(\text{Tåg} \vee \neg \text{Bomar}) \wedge$$

$$(\text{Bomar} \vee \text{Grön}) \wedge$$

$$(\text{Grön} \wedge (\neg \neg \text{Tåg} \vee \neg \neg \text{Bomar}))$$

$\stackrel{\text{taut}}{\iff}$

$$(\neg \text{Tåg} \vee \text{Röd}) \wedge$$

$$(\neg \text{Röd} \vee \text{Bil} \vee \text{Bomar}) \wedge$$

$$(\neg \text{Röd} \vee \neg \text{Bil} \vee \text{Bbp}) \wedge$$

$$(\text{Tåg} \vee \neg \text{Bomar}) \wedge$$

$$(\text{Bomar} \vee \text{Grön}) \wedge$$

$$\text{Grön} \wedge$$

$$(\text{Tåg} \vee \text{Bomar})$$

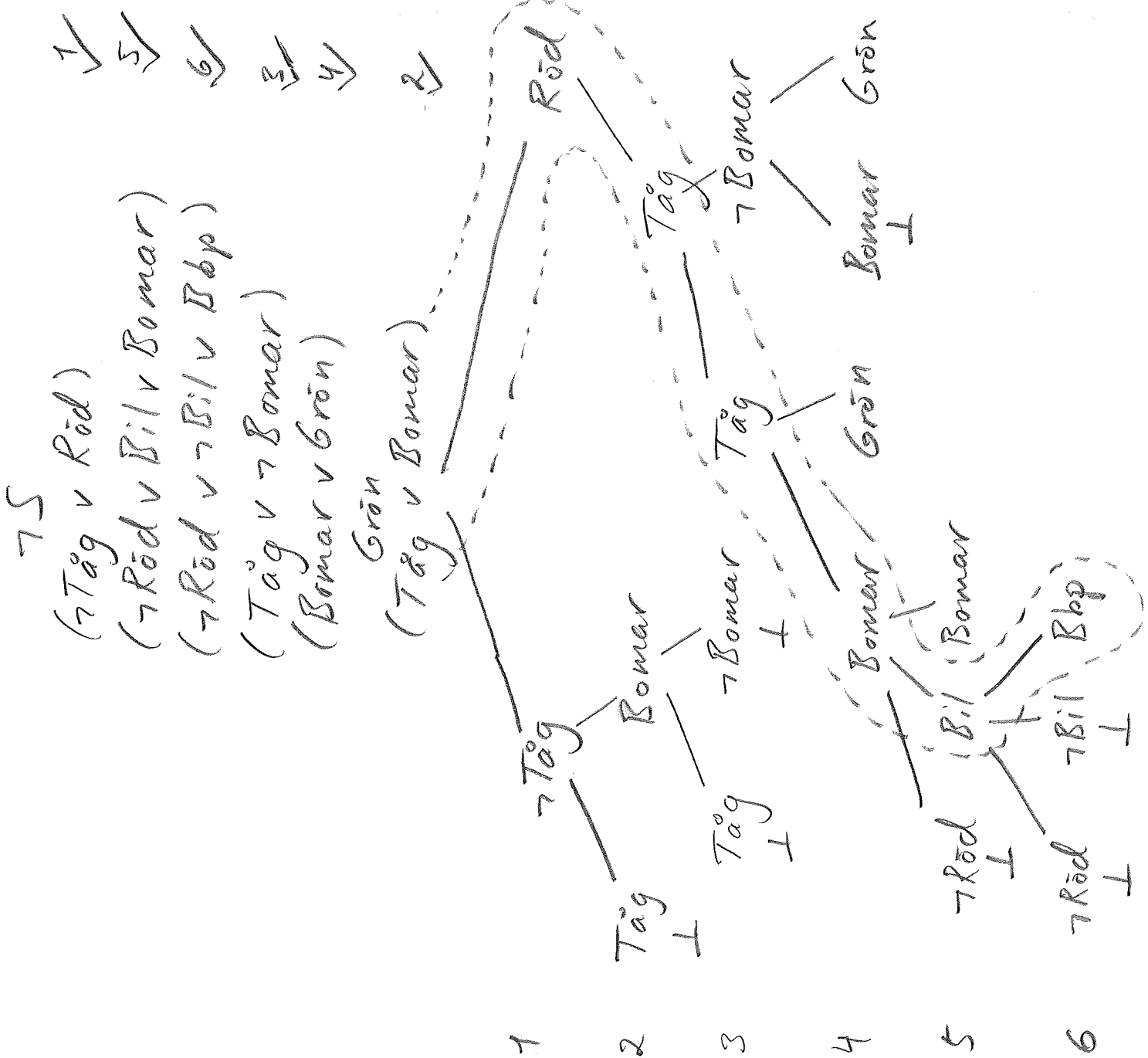
där den sista satsen är på KNF och
ekvivalent med $\neg S$.

⑥

När man har en sats på KNF så kan man ofta* få ett svar relativt snabbt på frågan om satsen är satisfierbar genom att använda "resolutionsmetoden" (vilket är datorernas favoritmetod) eller, som vi gör här, genom att använda semantiska träd (som jag tycker är enklare att hantera med "handräkning").

Vi börjar med den konjunktiva normalformen för $\neg S$ överst i trädet och eliminerar först all konjunktioner och får då

* men inte alltid



(8)

För stegen 5 och 6 är inte hela trädet ifyllt, men det behövs inte i detta fall för vi ser att det finns (minst) en gren utan motsägelser. En sådan gren är markerad och den ger upphov till sanningsvärdestilldelningen

$$Bbp = S$$

$$Bil = S$$

$$Bomar = S$$

$$Tåg = S$$

$$Röd = S$$

$$Grön = S$$

som satisfierar $\neg S$ och därmed är S inte en tautologi.

Så våra preliminära specifikationer har brister.

Antag att vi lägger till specifikationen

$$Röd \leftrightarrow \neg Grön$$

som är tautologiskt ekvivalent med

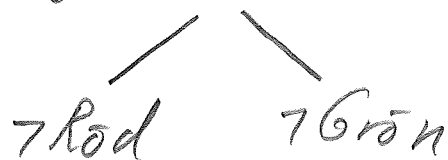
$$7. (\neg Röd \vee \neg Grön) \wedge (Röd \vee Grön)$$

Om nu T betecknar satsen

$$(1 \wedge 2 \wedge 3 \wedge 4 \wedge 5 \wedge 7) \rightarrow 6$$

så kommer den konjunktiva normalformen för $\neg T$ att se ut som den för $\neg S$ fortsett från att vi har lagt till 7 (som ytterligare konjunkt).

Utan att göra ett nytt semantiskt träd så kan man se att $\neg T$ inte är satisfierbar och därmed att T är en tautologi. För satsen 7 leder till att $\neg Röd \vee \neg Grön$ kommer att finnas med uppe i det nya trädet innan det har börjat förgrenas, och alla motsägelserfria grenar i det gamla trädet innehåller både Grön och Röd. Så när $\neg Röd \vee \neg Grön$ ger upphov till förgreningen (i det nya trädet)



så kommer varje gren som inte redan tidigare

var motsägelsetfull att innehålla någon av motsägelserna Röd, $\neg$ Röd eller Grön, $\neg$ Grön.

Så i det utvidgade nätet för $\neg T$ så blir alla grenar motsägelsetfulla, vilket betyder att $\neg T$ inte är satisfierbar.

Kommentar: Vi kan kolla om S och/eller T är tautologier med hjälp av sanningsvärdestabeller, men eftersom 6 satsvariabler är inblandade så kommer tabellen att ha $2^6 = 64$ rader. Med hjälp av semantiska träd så var problemet förfarande ganska lätt att lösa för hand.

Men i värsta fall kan det hända att även semantiska träd (och resolutionsmetoden) kräver "exponentiellt mycket resurser". Detta leder in på svåra problem inom komplexitetsteorin.