## O autômato produto

Mas, essa não é a única maneira de fazer as coisas.

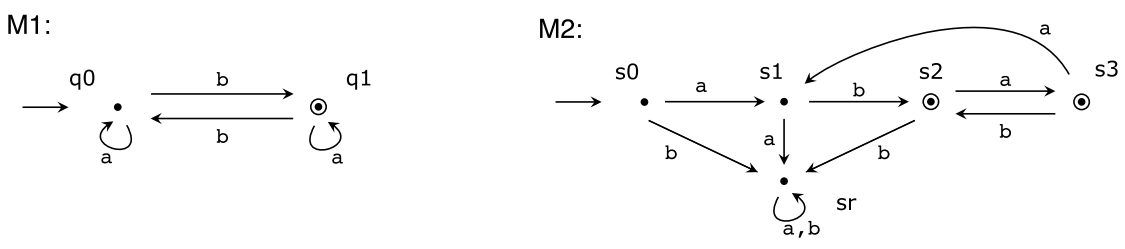
Quer dizer, a gente também pode raciocinar diretamente com autômatos.

Considere mais uma vez as linguagens

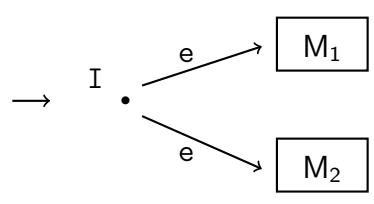
$L_1$ = palavras com um número ímpar de b's

$L_2$ = concatenações não-vazias de blocos ab e aba

E considere os autômatos $\mathsf{M}_1$ e $\mathsf{M}_2$ que reconhecem essas linguagens




E agora, imagine que nós construímos o seguinte autômato




*Mas, como é que esse negócio funciona?*

Bom, no começo da computação, o autômato passa do estado inicial I para $(\mathsf{q}_0, \mathsf{s}_0)$.

Daí, a gente lembra que $\mathsf{M}_1$ e $\mathsf{M}_2$ são autômatos determinísticos.

Isso significa que, a partir de agora, nós sempre vamos estar em um estado de $\mathsf{M}_1$ e um estado de $\mathsf{M}_2$.

Por exemplo,

$$\mathtt{I} \xrightarrow{\mathtt{e}} (\mathsf{q}_0, \mathsf{s}_0) \xrightarrow{\mathtt{a}} (\mathsf{q}_0, \mathsf{s}_1) \xrightarrow{\mathtt{b}} (\mathsf{q}_1, \mathsf{s}_2) \xrightarrow{\mathtt{a}} (\mathsf{q}_1, \mathsf{s}_3) \longrightarrow \cdots$$

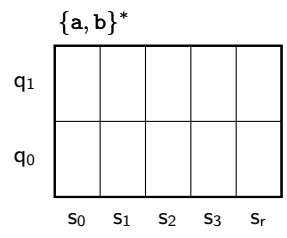
Quer dizer, o que nós temos aqui é uma máquina que executa $\mathsf{M}_1$ e $\mathsf{M}_2$ em paralelo.

*Mas, o que é que essa máquina faz?*

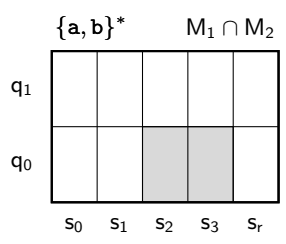
Bom, aqui nós podemos raciocinar em termos de linguagens.

Quer dizer, os diagramas de $\mathsf{M}_1$ e $\mathsf{M}_2$ particionam o conjunto de todas as palavras assim

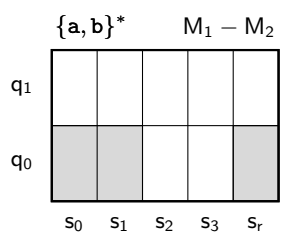

Daí que, quando nós executamos $\mathsf{M}_1$ e $\mathsf{M}_2$ em paralelo, nós obtemos a seguinte partição




Finalmente, se a gente quer o autômato que reconhece $\mathsf{L}_1 \cap \mathsf{L}_2$, basta marcar os pares que contém estados finais de $\mathsf{M}_1$ e $\mathsf{M}_2$ como finais




Por outro lado, se a gente quiser um autômato que reconhece a linguagem $\mathsf{L}_1 - \mathsf{L}_2$, basta marcar os estados finais assim




*Legal, não é?*

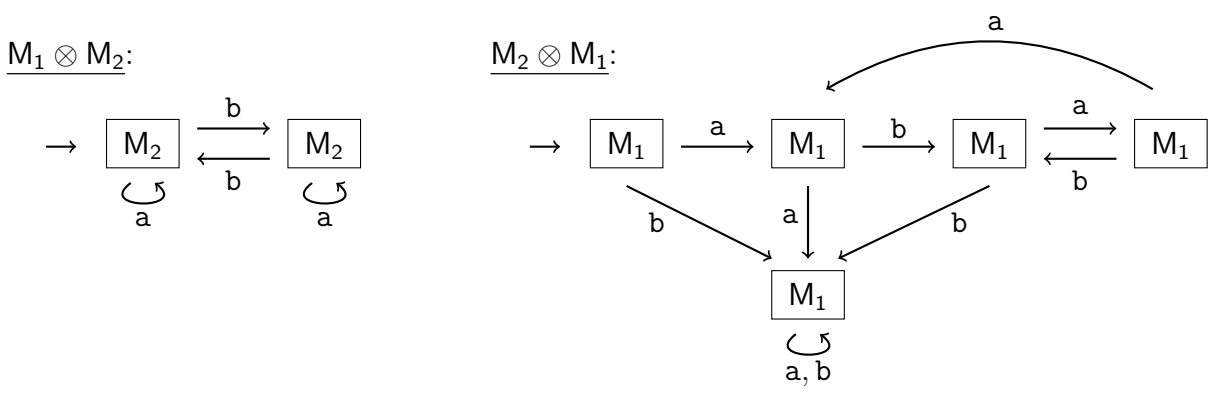
Sim, mas tem uma coisa ainda mais legal.

Quer dizer, imagine que nós queremos um autômato determinístico que reconhece a linguagem $\mathsf{L}_1 \cap \mathsf{L}_2$.

Daí, uma ideia seria aplicar o método sistemático da aula 09.

Mas, nós também podemos construir o *autômato produto* de $\mathsf{M}_1$ e $\mathsf{M}_2$.

Quer dizer, existem duas maneiras de fazer isso




A ideia é que nós substituímos os estados de um autômato por cópias do outro autômato.

E agora, nós podemos expadir o primeiro deles assim