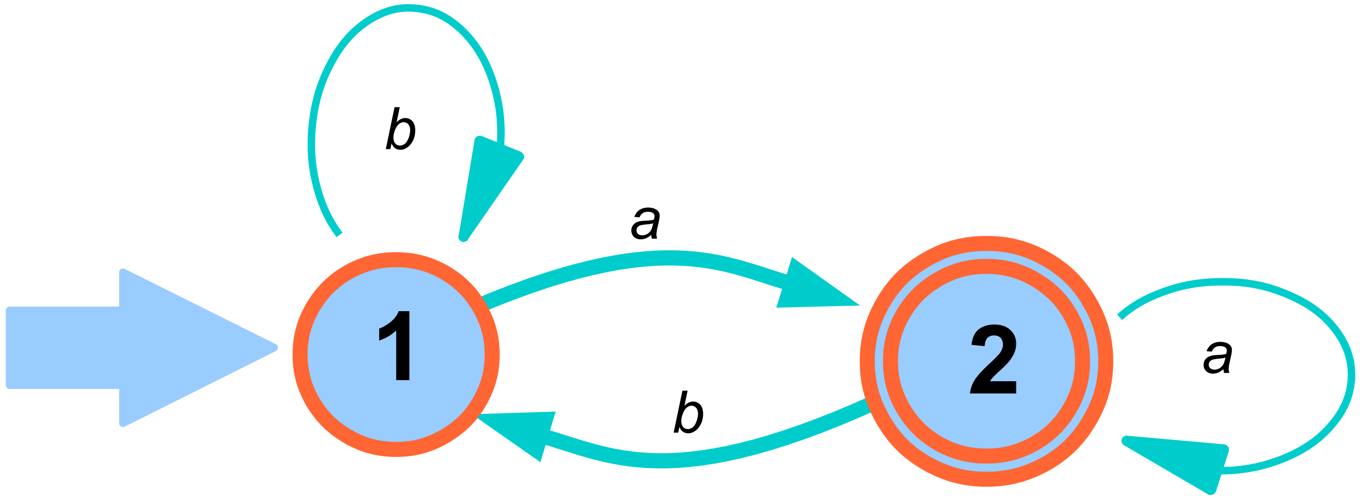
Opérations entre automates finis déterministes

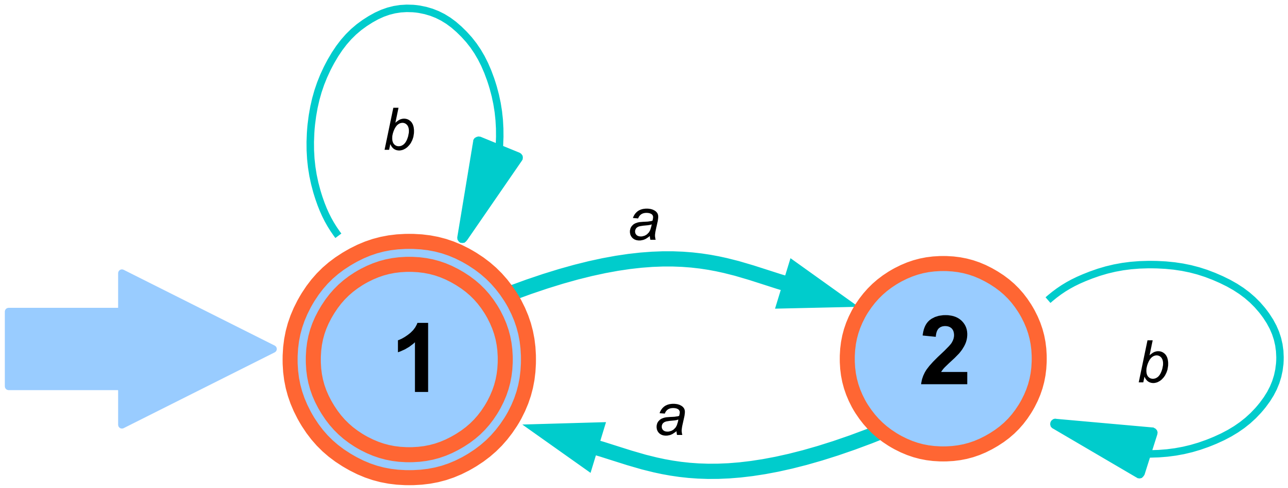
Si $\mathcal{A}$ et $\mathcal{A}'$ sont des automates finis déterministes respectivement à n et p états, reconnaissant les langages L et L', nous allons voir comment construire des automates (pas toujours déterministes) reconnaissant :

- $\overline{L}$, le langage complémentaire de L,
- la somme des langages L+L'
- l'intersection $L \cap L'$
- la différence $L \setminus L'$
- le langage L+
- et enfin le langage étoile L*

Nous illustrerons ces opérations à l'aide des deux automates suivant :



$\mathcal{A}$ reconnaissant les mots qui finissent par « a »



$\mathcal{A}'$ reconnaissant les mots ayant un nombre pair de « a »

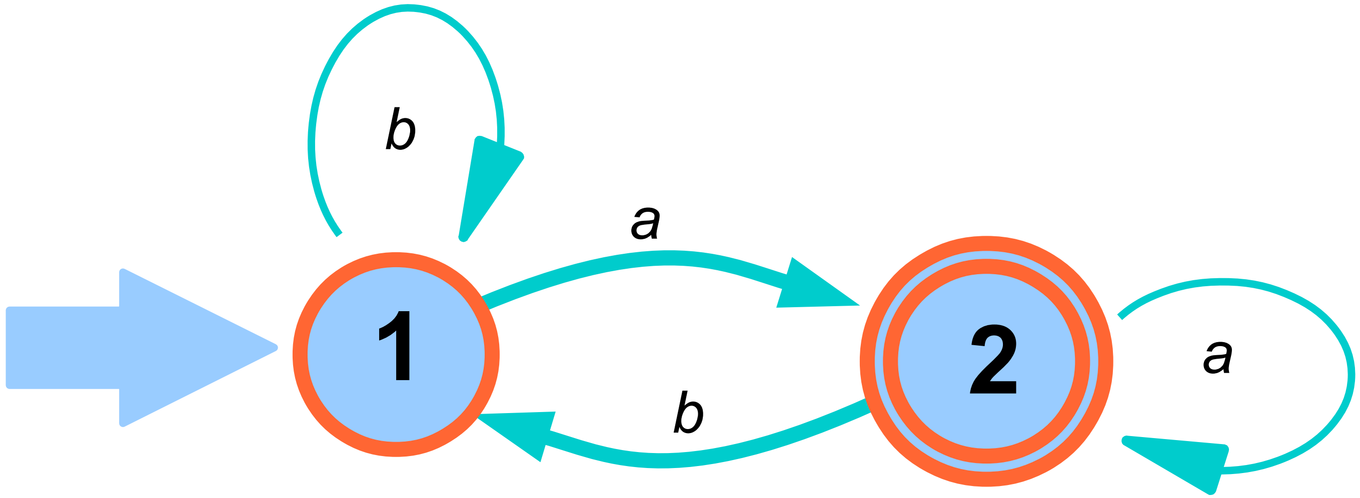
Complémentaire

Si l'automate est complet, il suffit d'inverser les états acceptants et refusants :

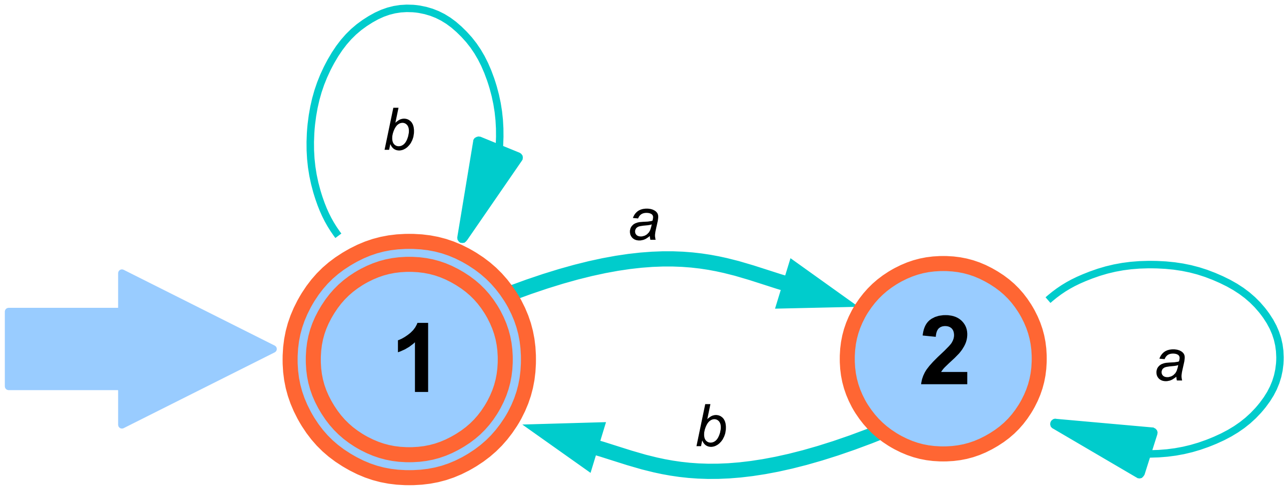
1. Compléter l'automate s'il n'est pas complet
2. Rendre acceptant les états refusant et réciproquement

On obtient un automate déterministe à n états

Complémentaire :



𝒜 reconnaissant les mots qui finissent par « a »



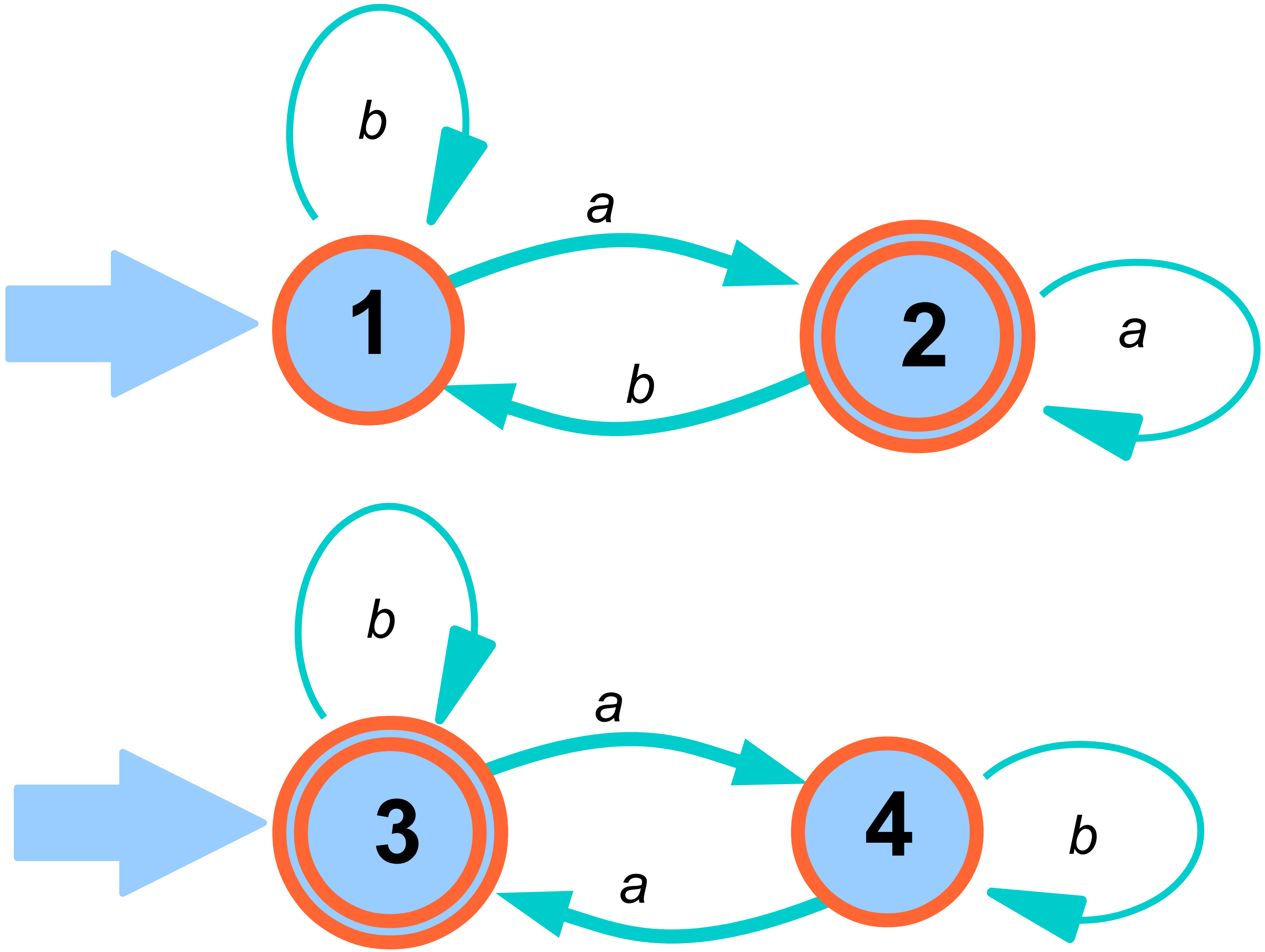
Son complémentaire reconnaissant les mots qui ne finissent pas par « a »

Somme

Il suffit de juxtaposer les deux automates (en renommant les états) et d'autoriser les deux états initiaux.

L'automate obtenu possède n+p états et n'est pas déterministe puisqu'il possède deux états initiaux.

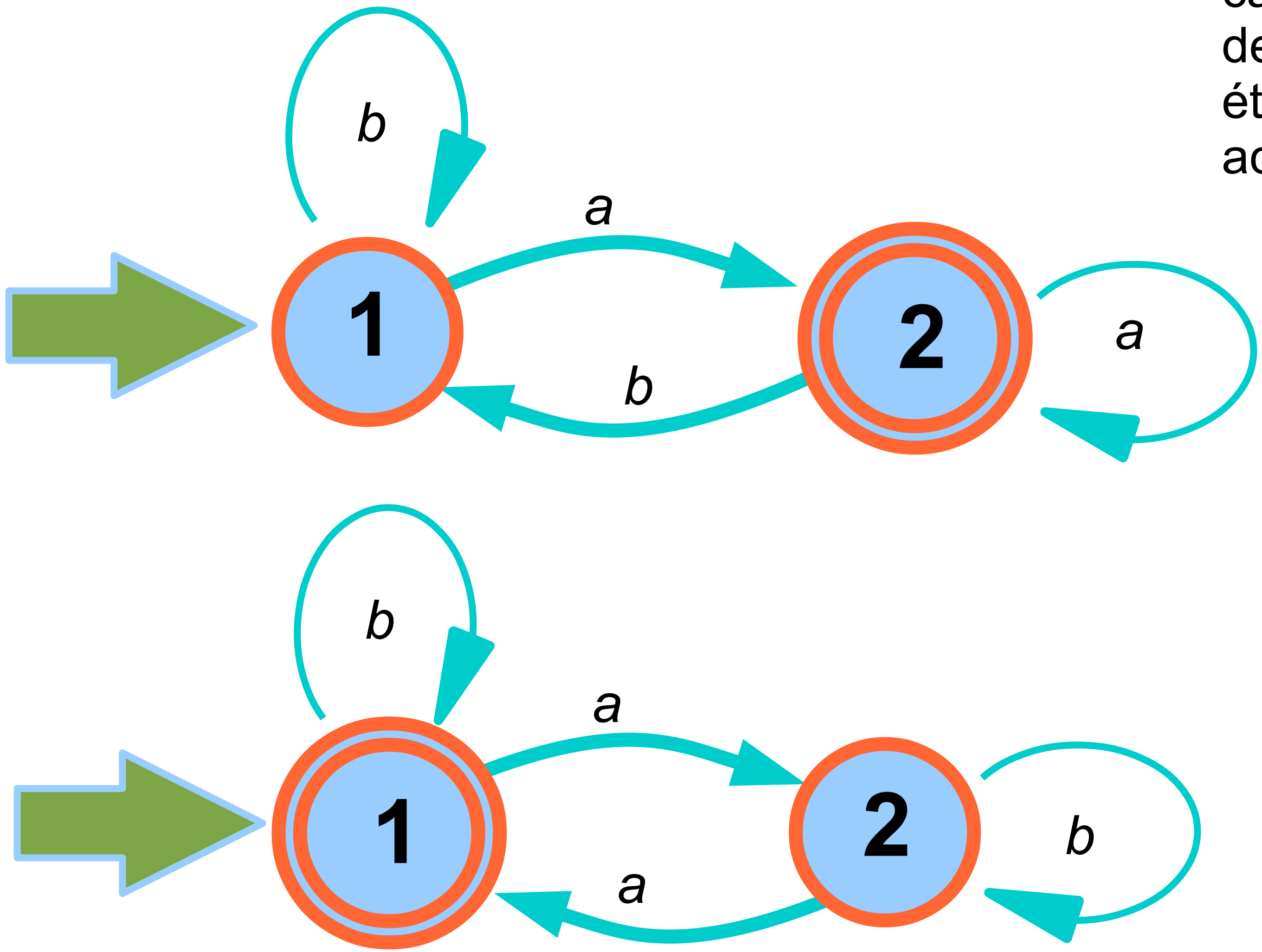
Somme



- En fonction de l'état initial choisi, on va reconnaître les mots finissant par « a » ou « ayant un nombre pair de « a ».
- L'automate reconnaît bien L+L'

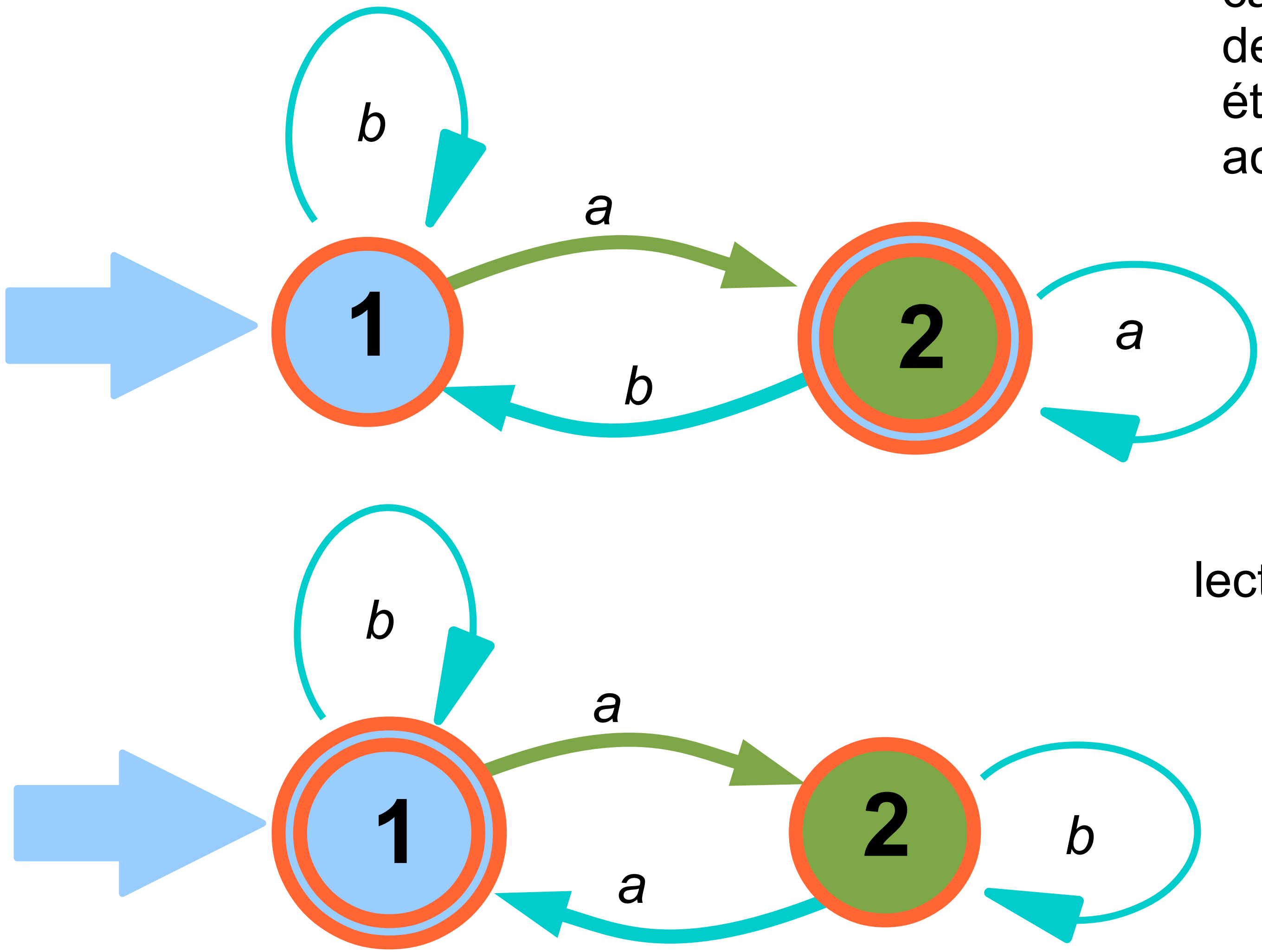
Intersection

Le mot « abba » appartient à l'intersection : $L \cap L'$
car la lecture conjointe des deux automates conduit des états initiaux à deux états acceptants



Intersection

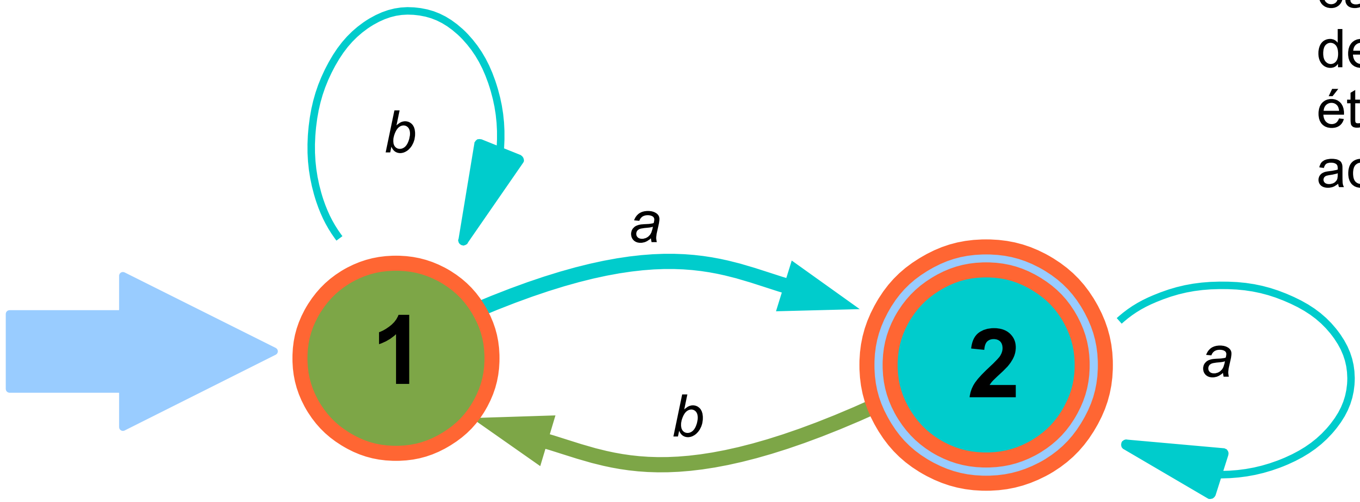
Le mot « abba » appartient à l'intersection : $L \cap L'$
car la lecture conjointe des deux automates conduit des états initiaux à deux états acceptants



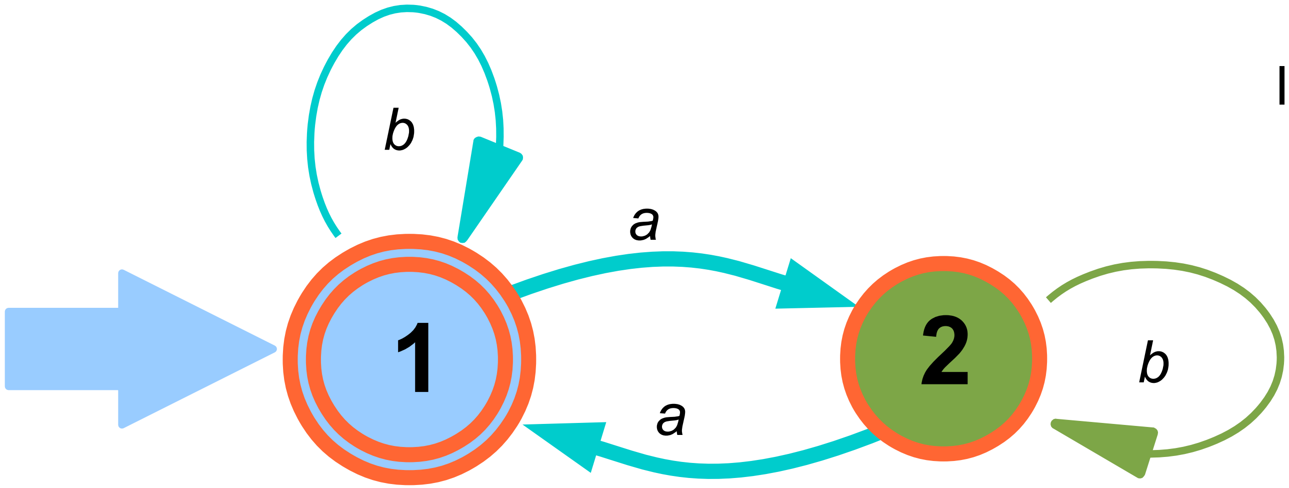
lecture de « a »

Intersection

Le mot « abba » appartient à l'intersection : $L \cap L'$
car la lecture conjointe des deux automates conduit des états initiaux à deux états acceptants

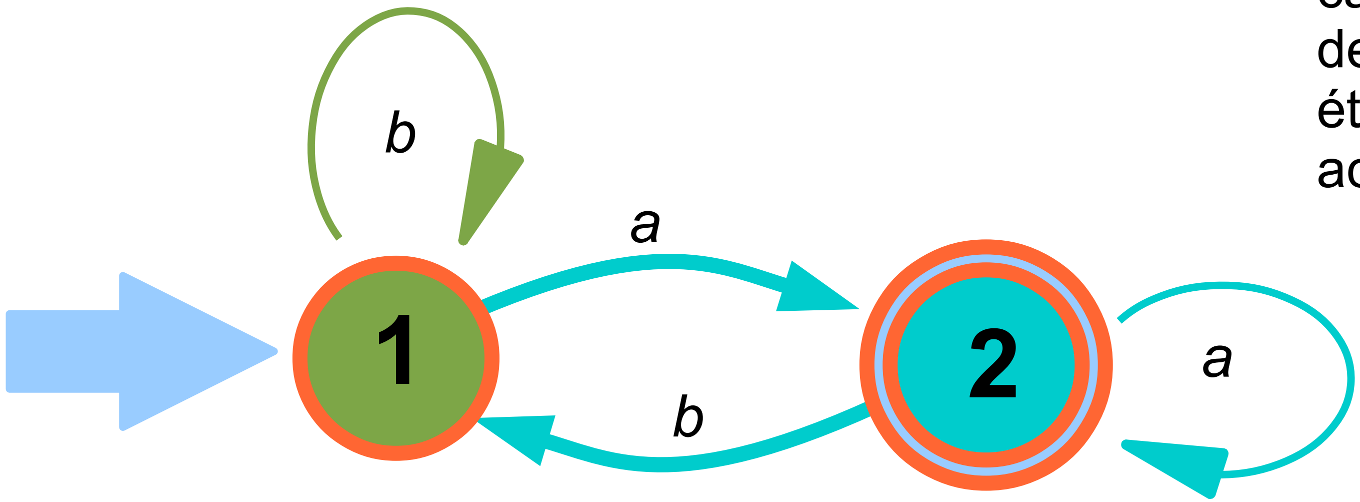


lecture de « ab »

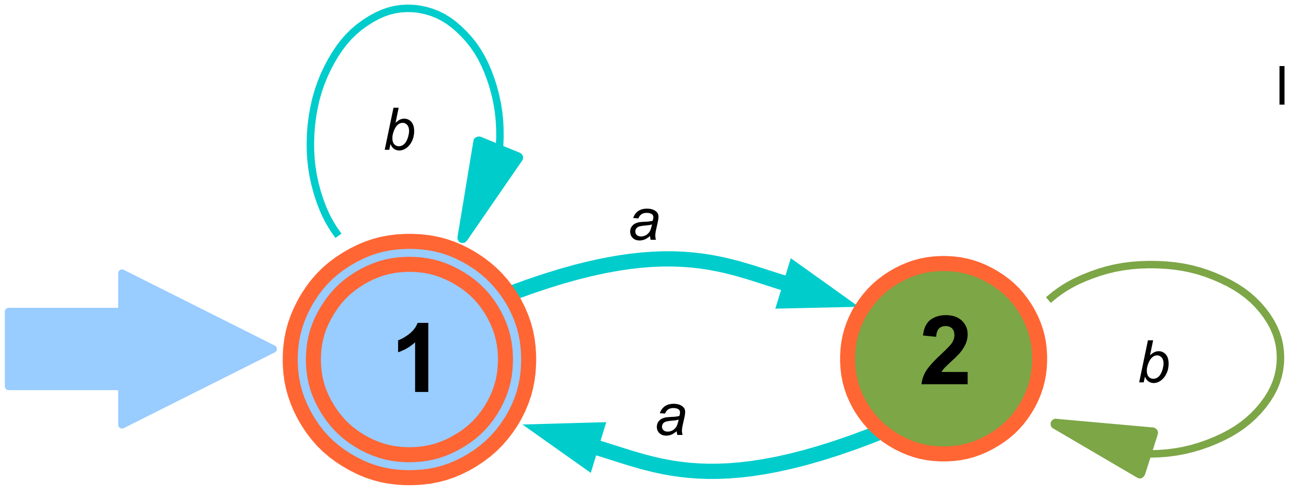


Intersection

Le mot « abba » appartient à l'intersection : L ∩ L'
car la lecture conjointe des deux automates conduit des états initiaux à deux états acceptants

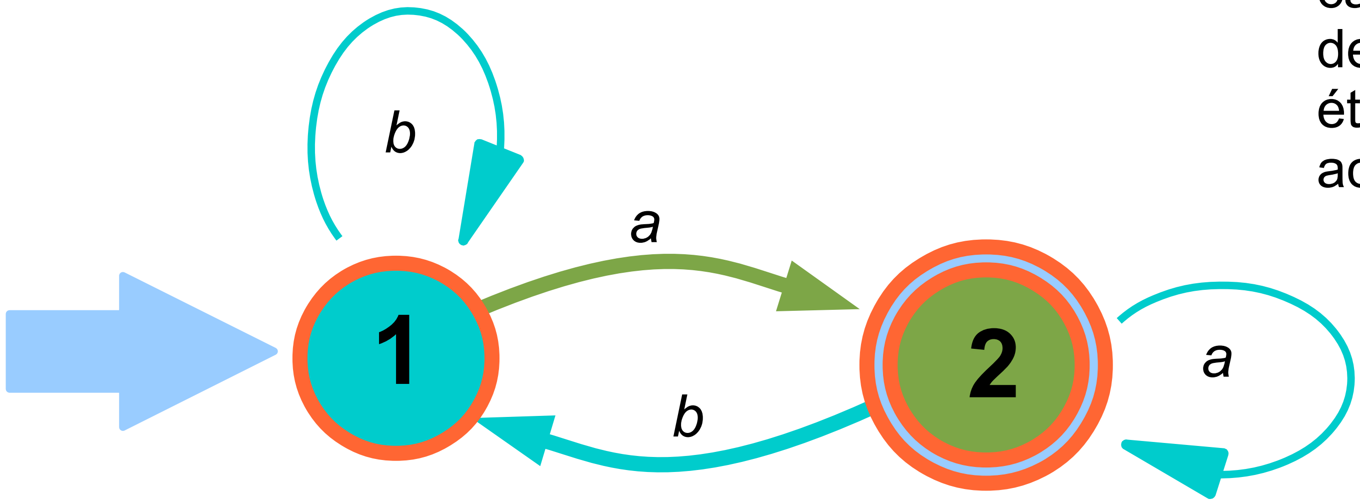


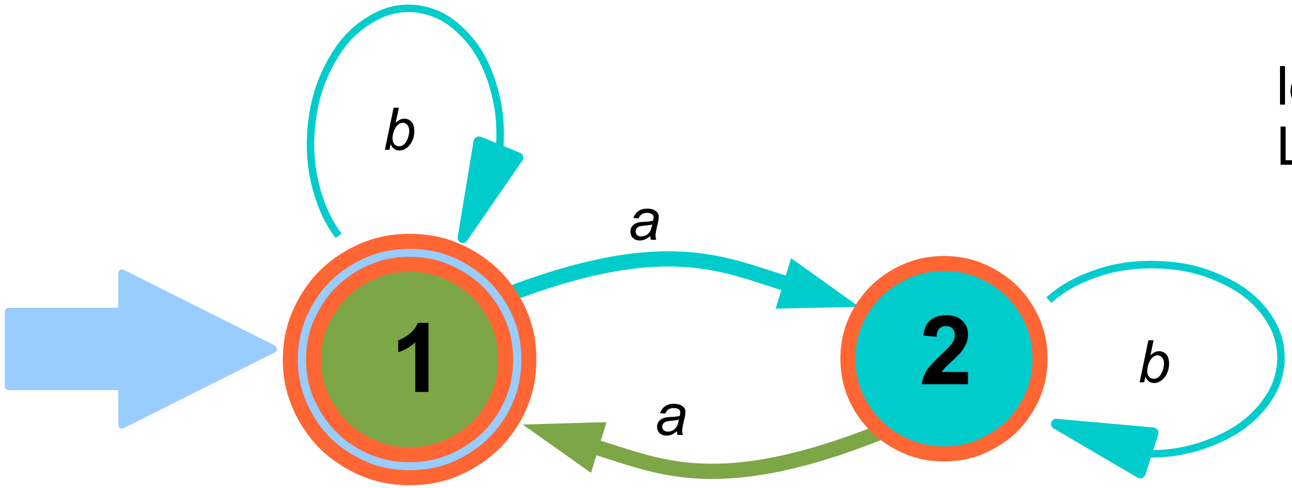
lecture de « abb »



Intersection

Le mot « abba » appartient à l'intersection : $L \cap L'$
car la lecture conjointe des deux automates conduit des états initiaux à deux états acceptants





lecture de « abba »
Le mot est accepté.

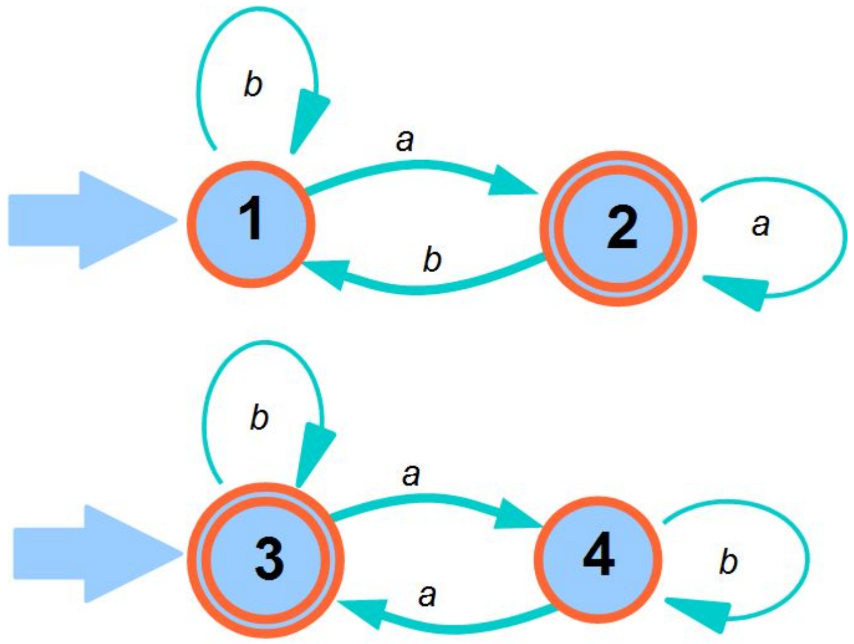
Intersection

Pour effectuer les deux lectures conjointement, il suffit de considérer l'automate des couples :

- En partant du couple des états initiaux, on effectue les transitions par couples.

- Un état est acceptant s''il est constitué d'un couple d'états acceptants.

Intersection

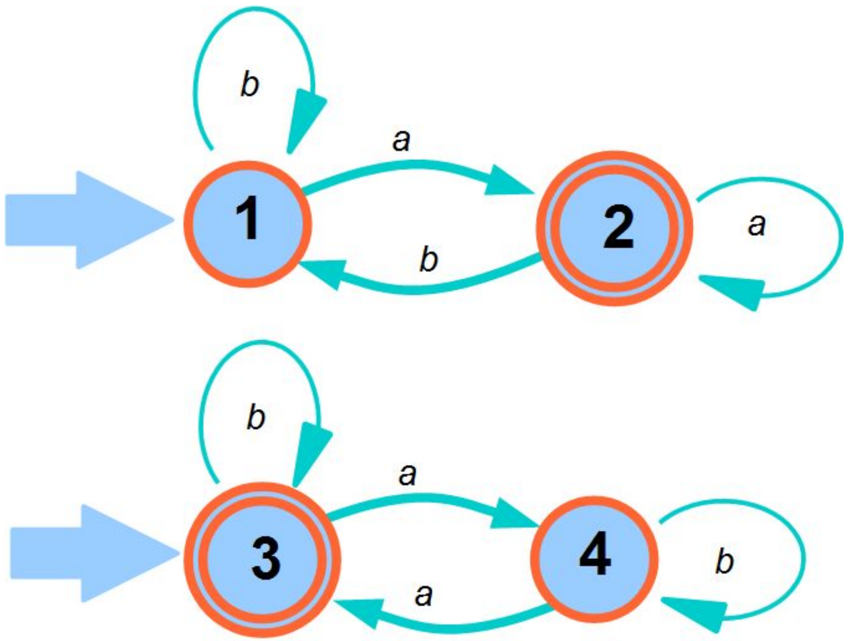


Partant du couple des états initiaux, on effectue les transitions par couples.

Q \ Σ	a	b
I = {1,3}		

Intersection

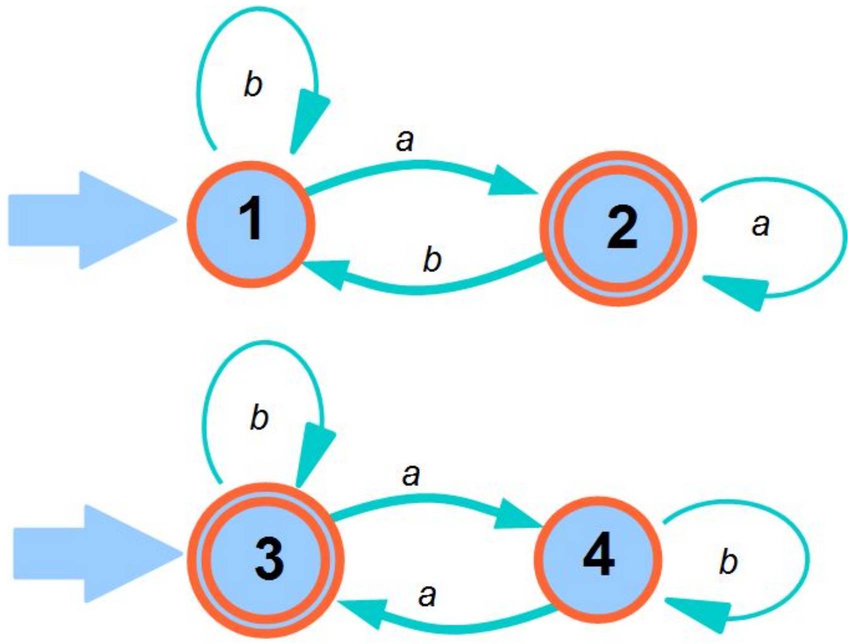
Partant du couple des états initiaux, on effectue les transitions par couples.



Q \ Σ	a	b
I = {1,3}	II = {2,4}	I

Intersection

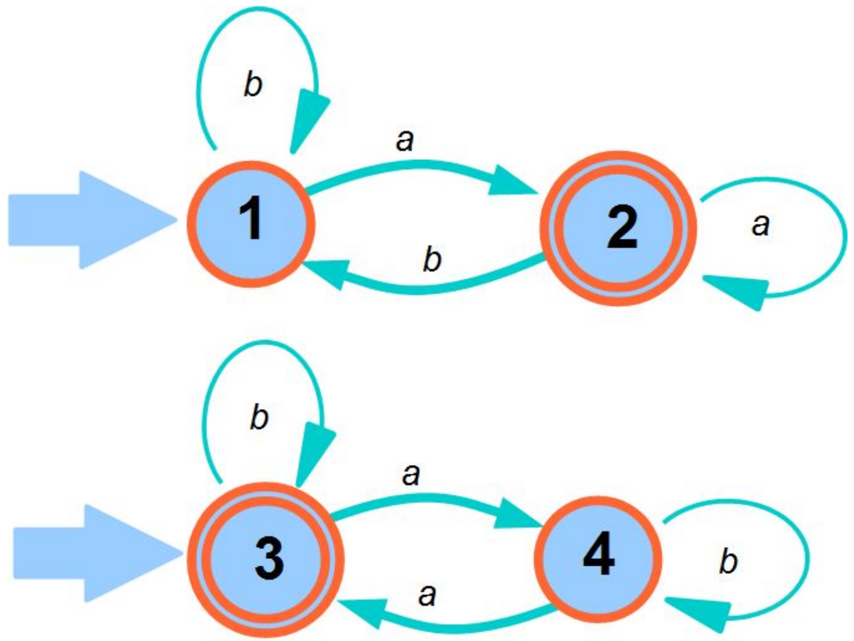
Partant du couple des états initiaux, on effectue les transitions par couples.



Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}

Intersection

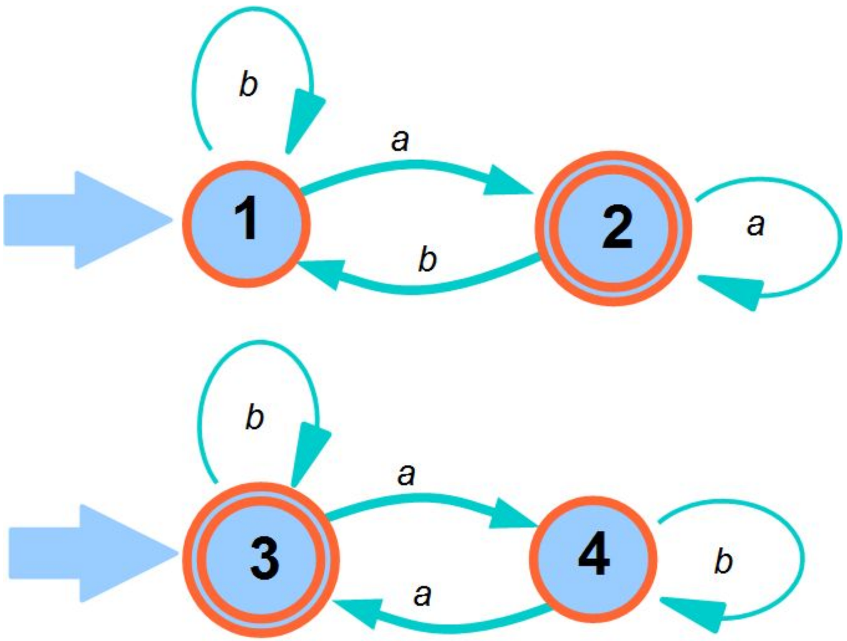
Partant du couple des états initiaux, on effectue les transitions par couples.



Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}
III = {2,3}	II	I

Intersection

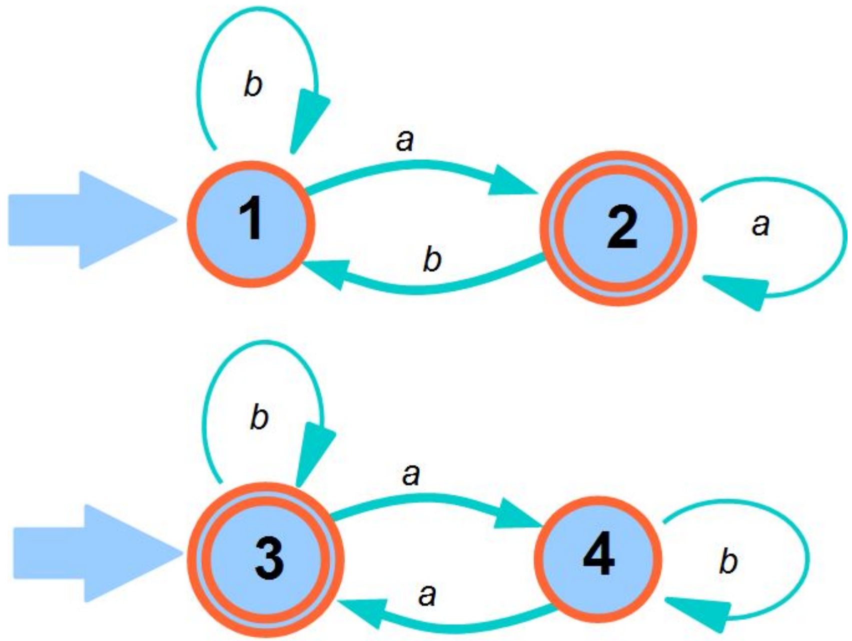
Partant du couple des états initiaux, on effectue les transitions par couples.



Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}
III = {2,3}	II	I
IV = {1,4}	III	IV

Intersection

Partant du couple des états initiaux, on effectue les transitions par couples.

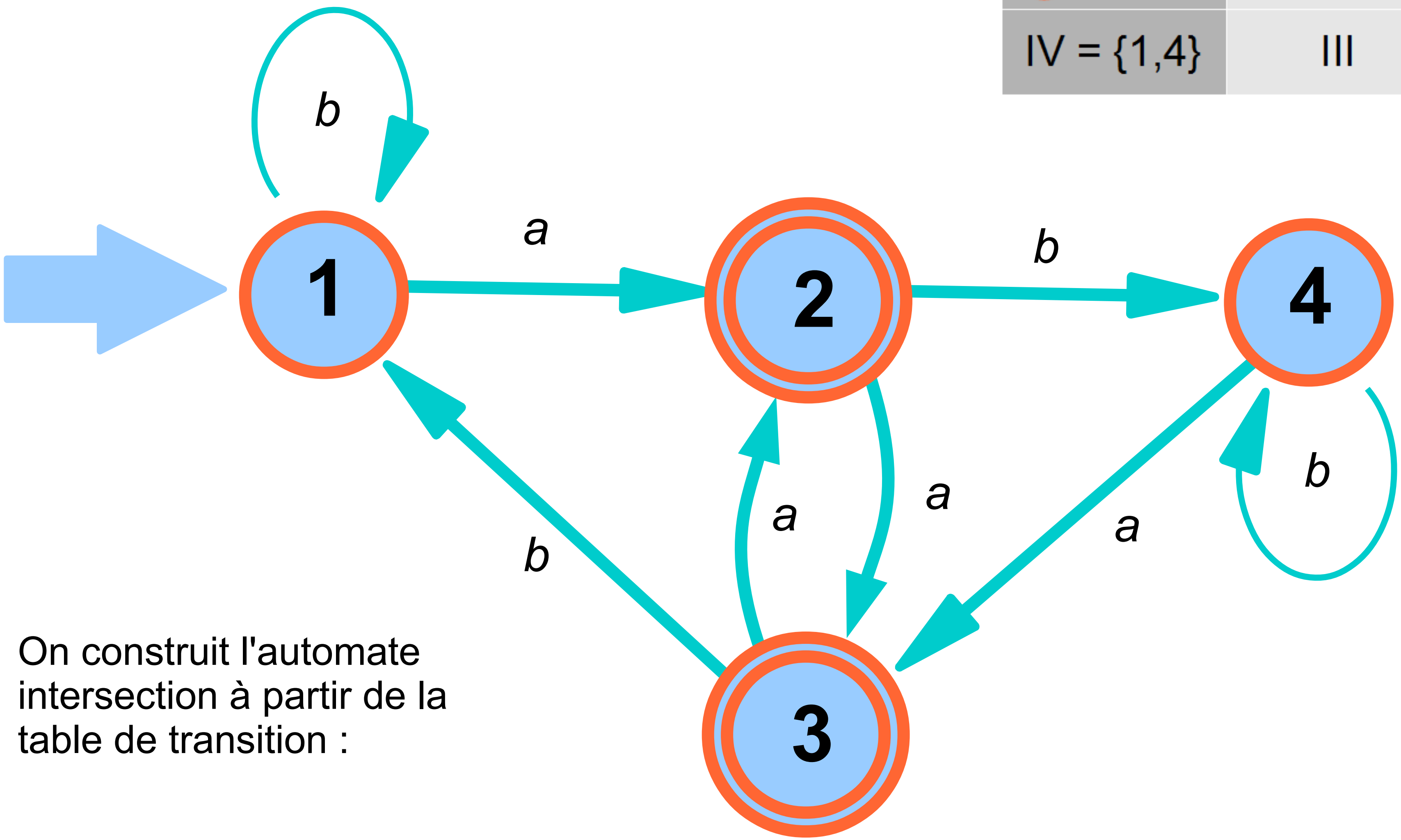


Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}
III = {2,3}	II	I
IV = {1,4}	III	IV

L'état {2,3} est l'unique état acceptant

Intersection

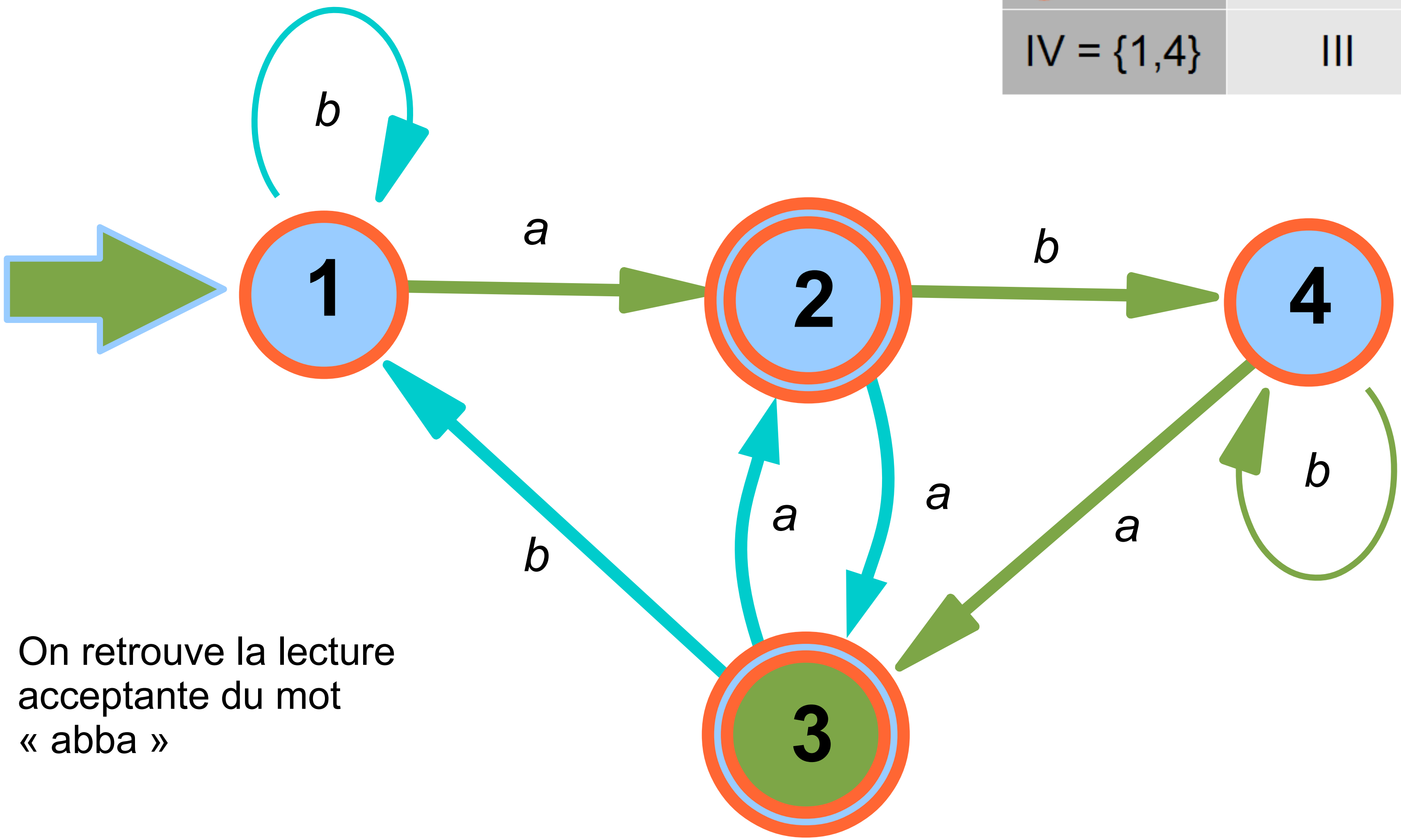
Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}
III = {2,3}	II	I
IV = {1,4}	III	IV



On construit l'automate intersection à partir de la table de transition :

Intersection

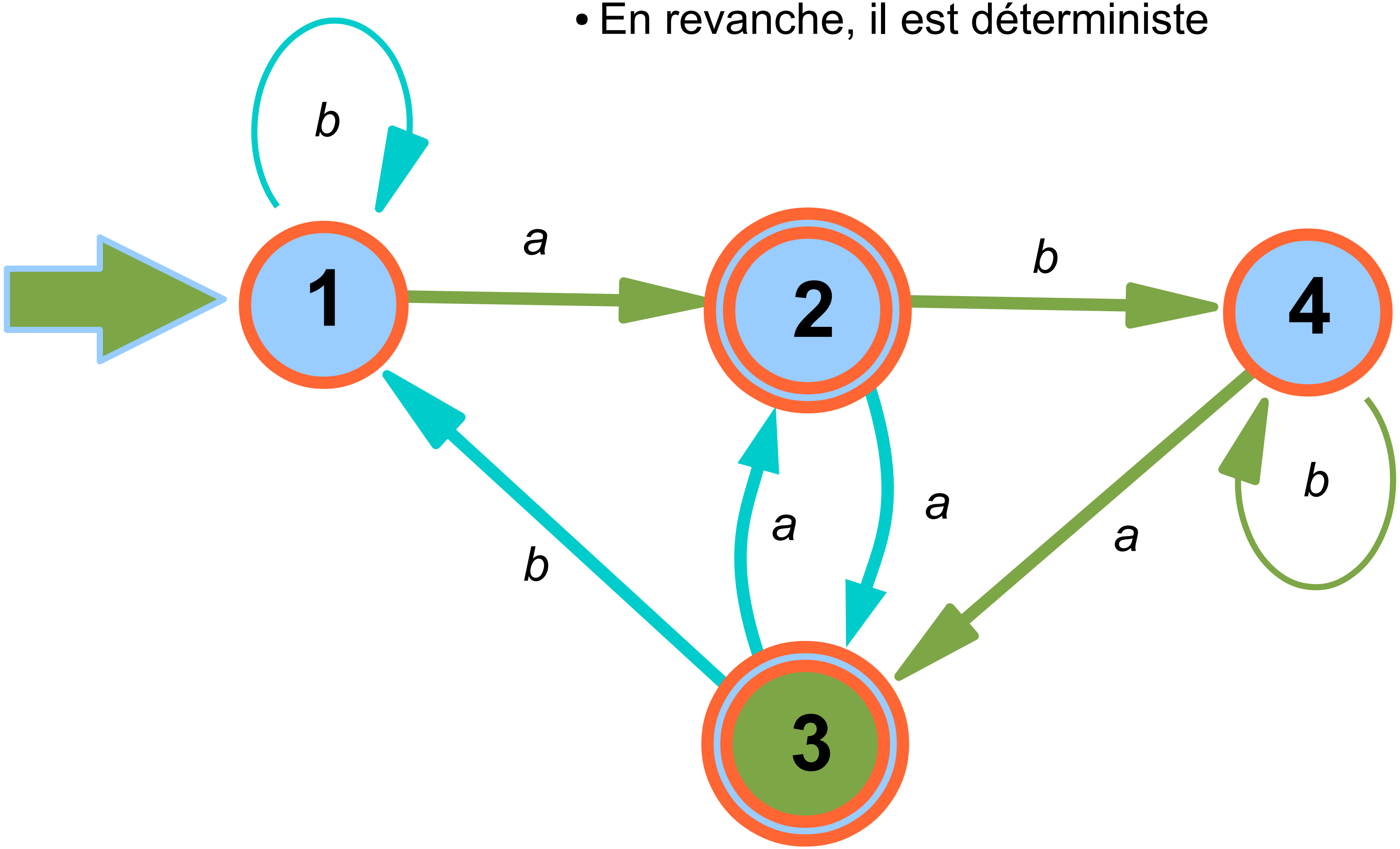
Q \ Σ	a	b
I = {1,3}	II = {2,4}	I
II = {2,4}	III = {2,3}	IV = {1,4}
III = {2,3}	II	I
IV = {1,4}	III	IV



On retrouve la lecture acceptante du mot « abba »

Intersection

- L'automate obtenu peut posséder jusqu'à n.p états
- En revanche, il est déterministe



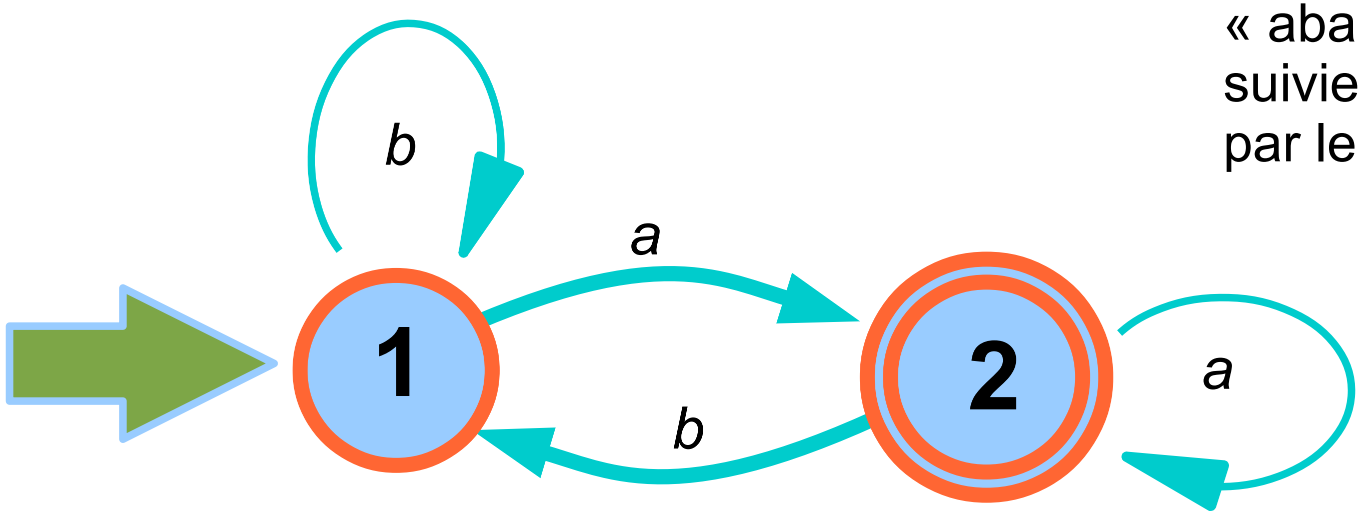
Différence

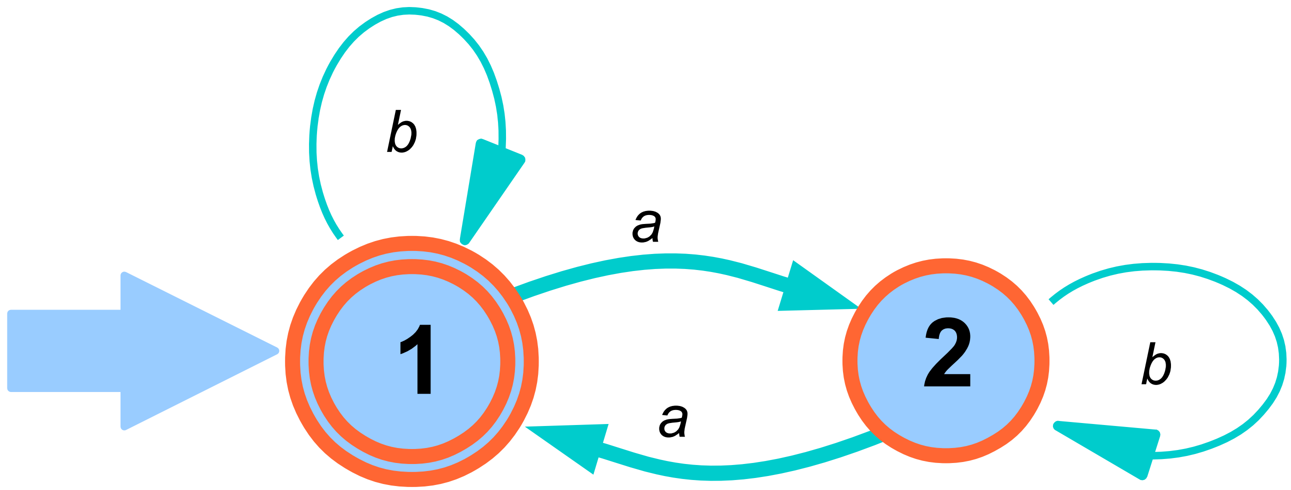
La différence de langages L \ L' constituée des mots de L qui n'appartiennent pas à L' est aussi égale à $L \cap \overline{L'}$

- On l'obtient donc à l'aide d'un calcul de complémentaire et d'une intersection.

Produit

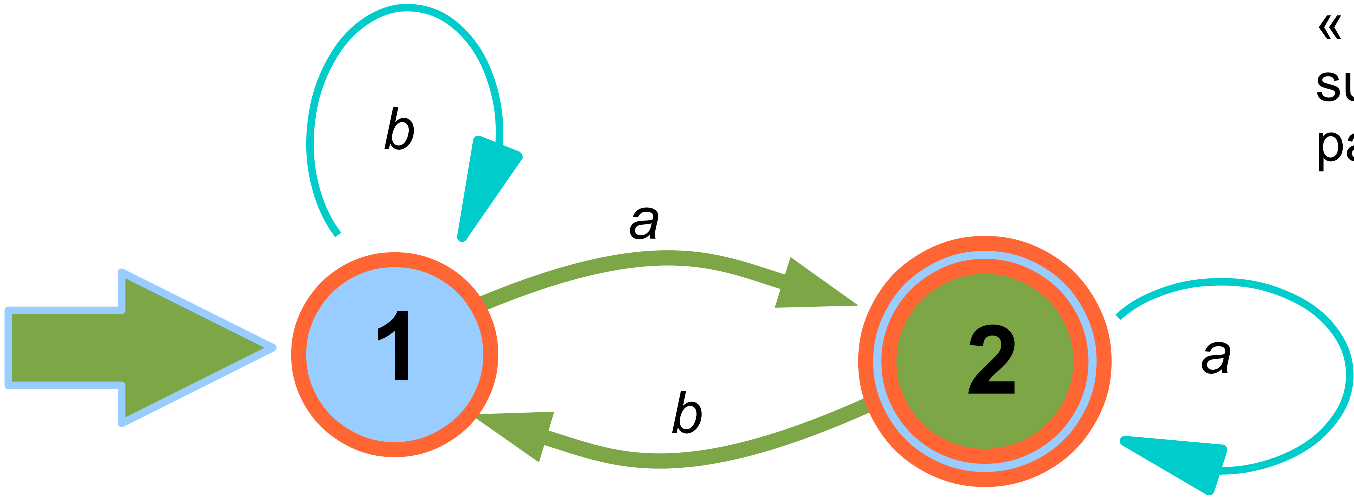
Le mot « aba.aab » appartient au produit L. L' car on peut le lire en enchaînant la lecture de « aba » par le premier automate suivie de la lecture de « aab » par le deuxième.



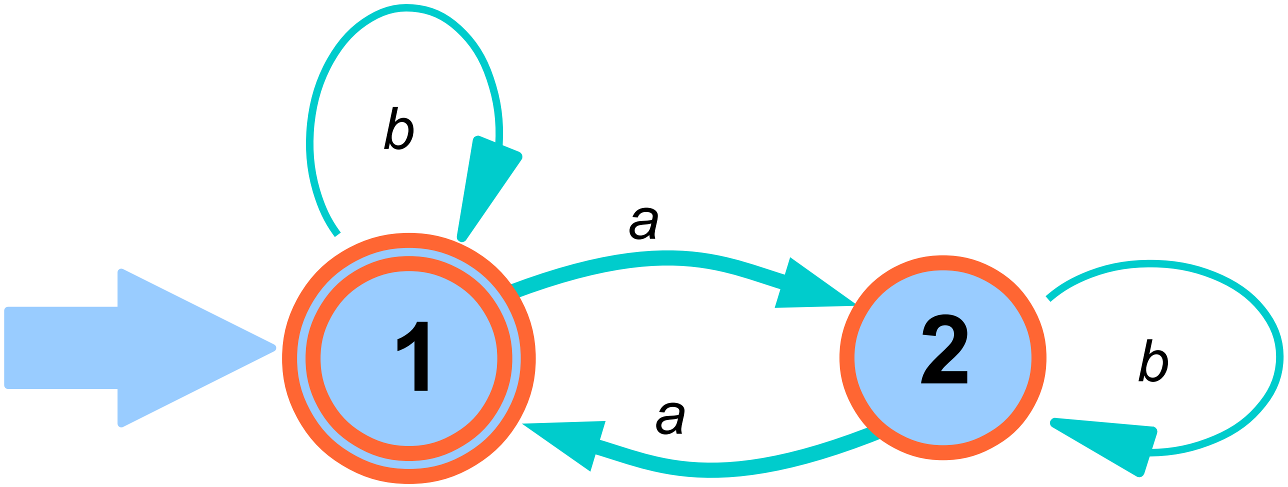


Produit

Le mot « aba.aab » appartient au produit L. L' car on peut le lire en enchaînant la lecture de « aba » par le premier automate suivie de la lecture de « aab » par le deuxième.

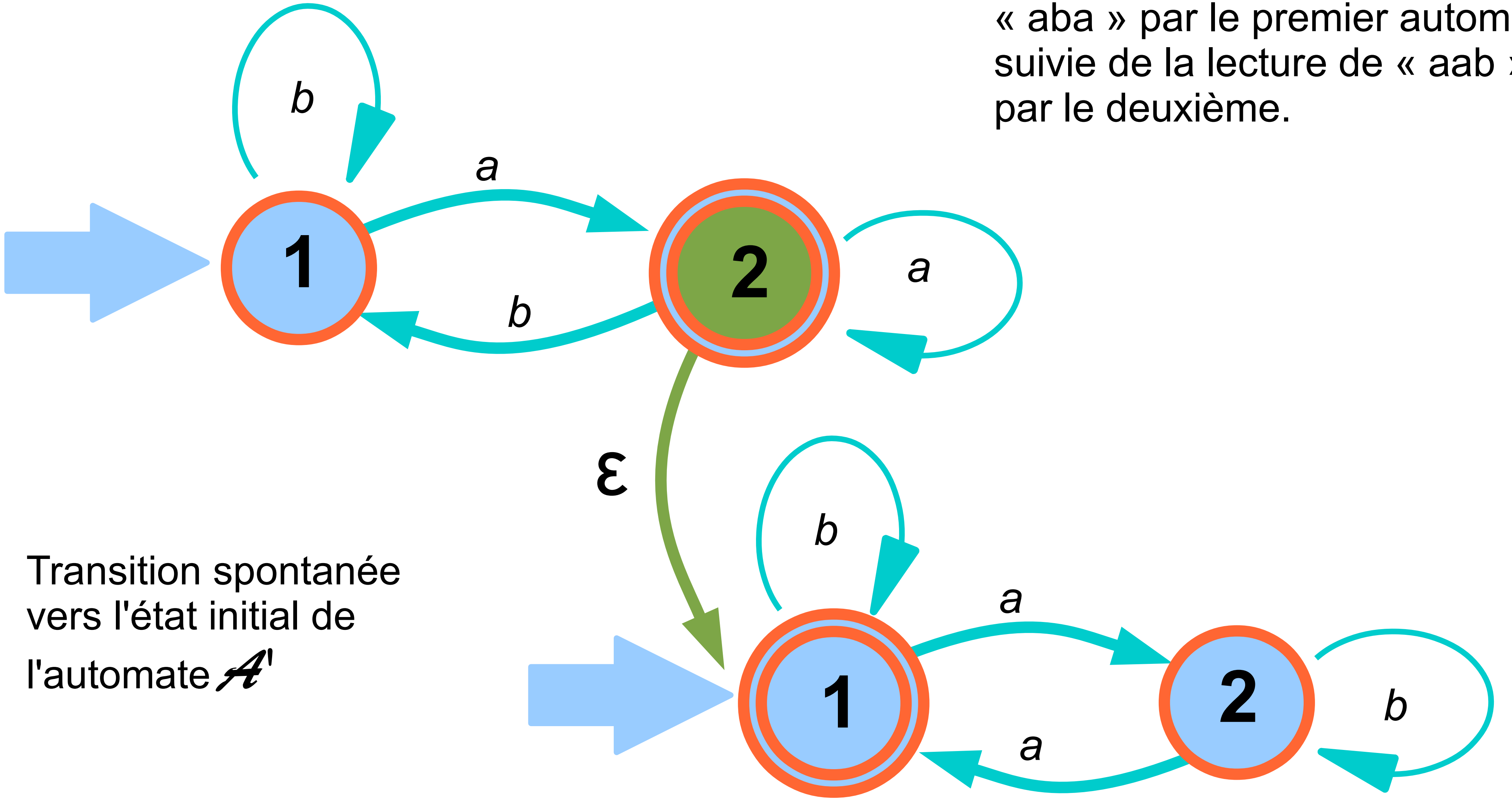


lecture acceptante de « aba » par l'automate $\mathcal{A}$



Produit

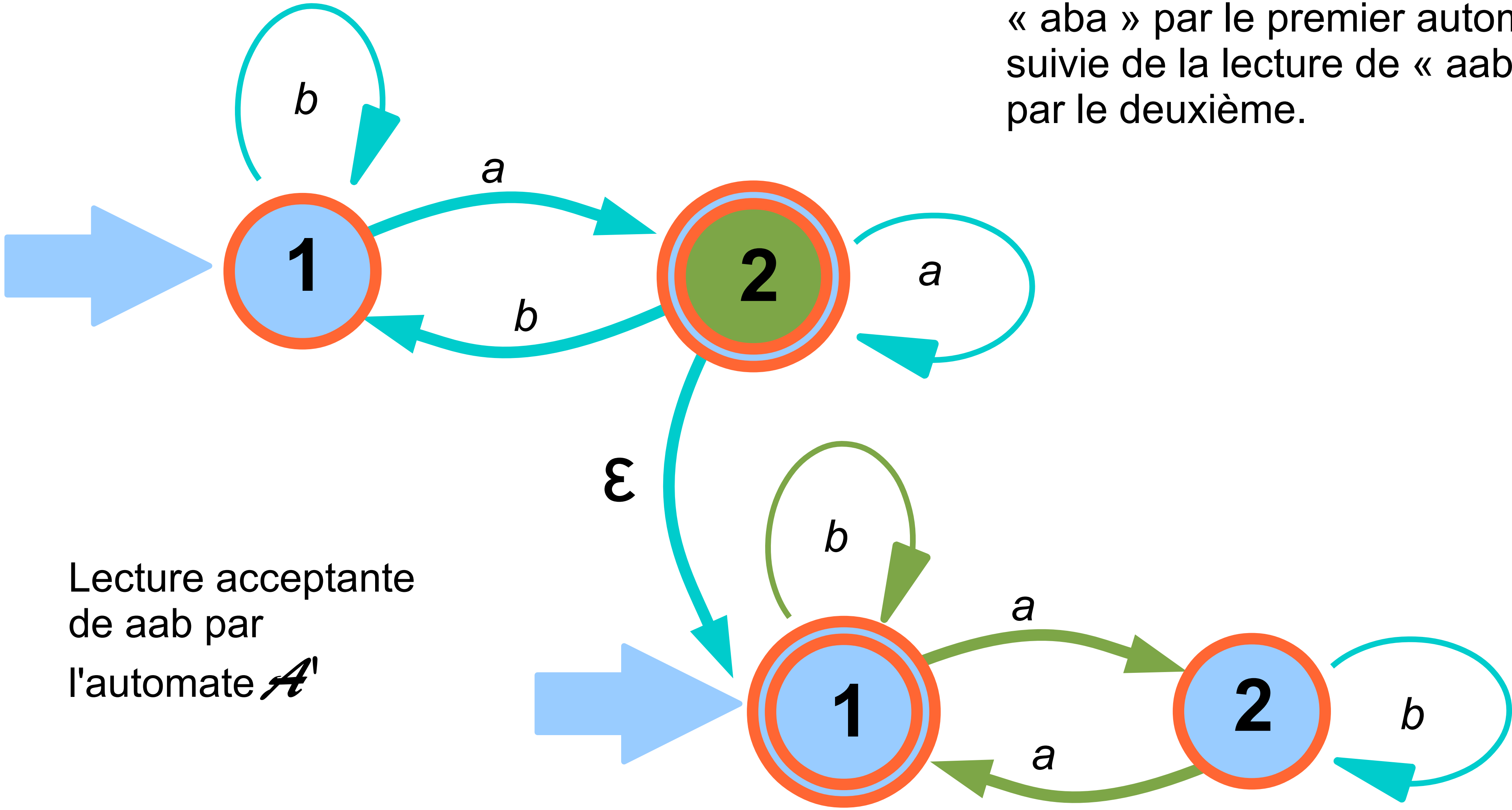
Le mot « aba.aab » appartient au produit L. L' car on peut le lire en enchaînant la lecture de « aba » par le premier automate suivie de la lecture de « aab » par le deuxième.



Transition spontanée vers l'état initial de l'automate $\mathcal{A}'$

Produit

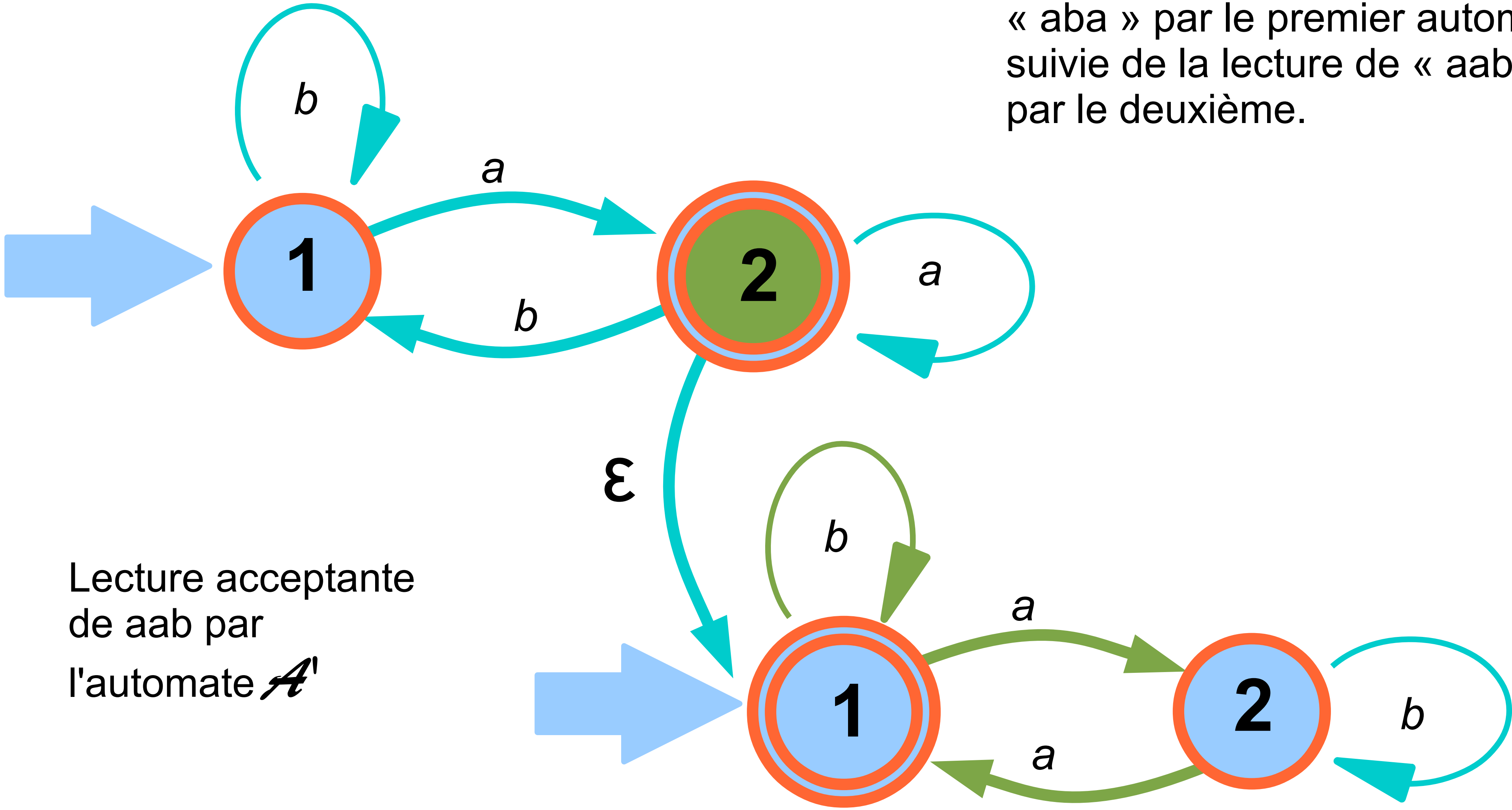
Le mot « aba.aab » appartient au produit L. L' car on peut le lire en enchaînant la lecture de « aba » par le premier automate suivie de la lecture de « aab » par le deuxième.



Lecture acceptante de aab par l'automate $\mathscr{A}'$

Produit

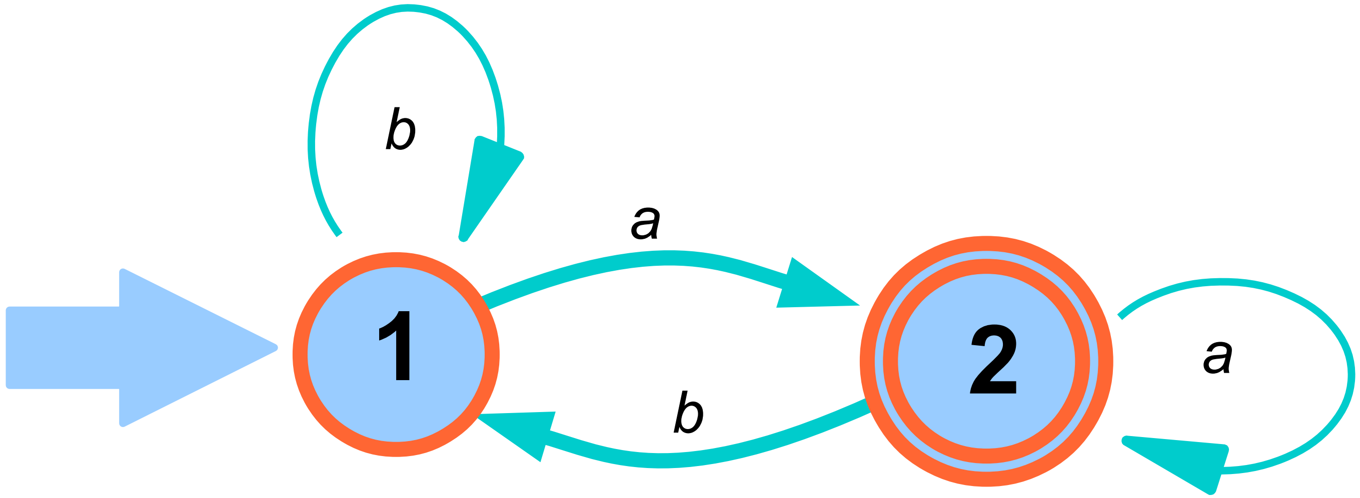
Le mot « aba.aab » appartient au produit L. L' car on peut le lire en enchaînant la lecture de « aba » par le premier automate suivie de la lecture de « aab » par le deuxième.

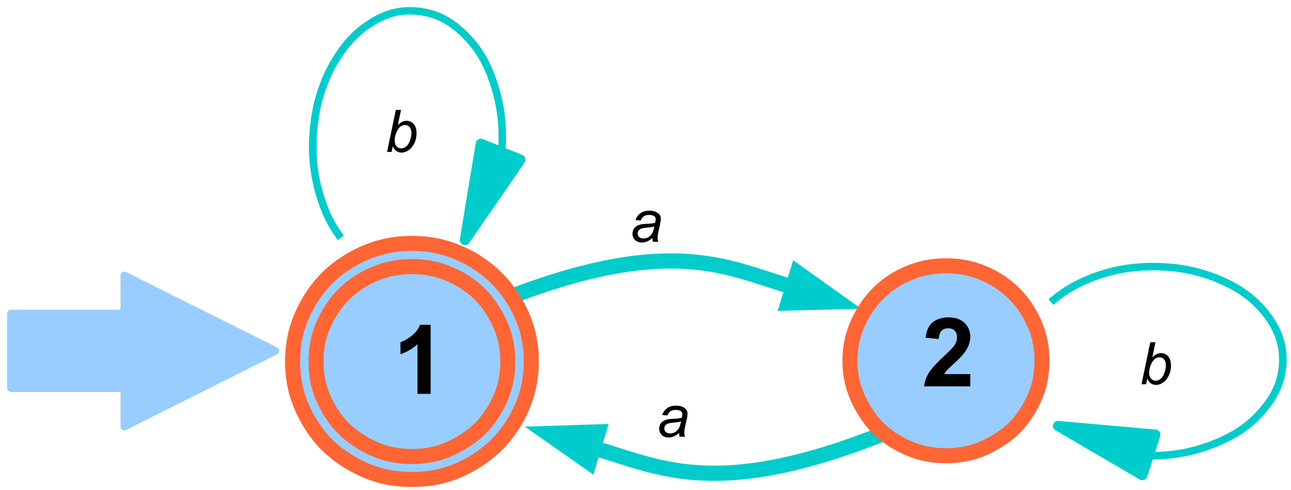


Lecture acceptante de aab par l'automate $\mathcal{A}'$

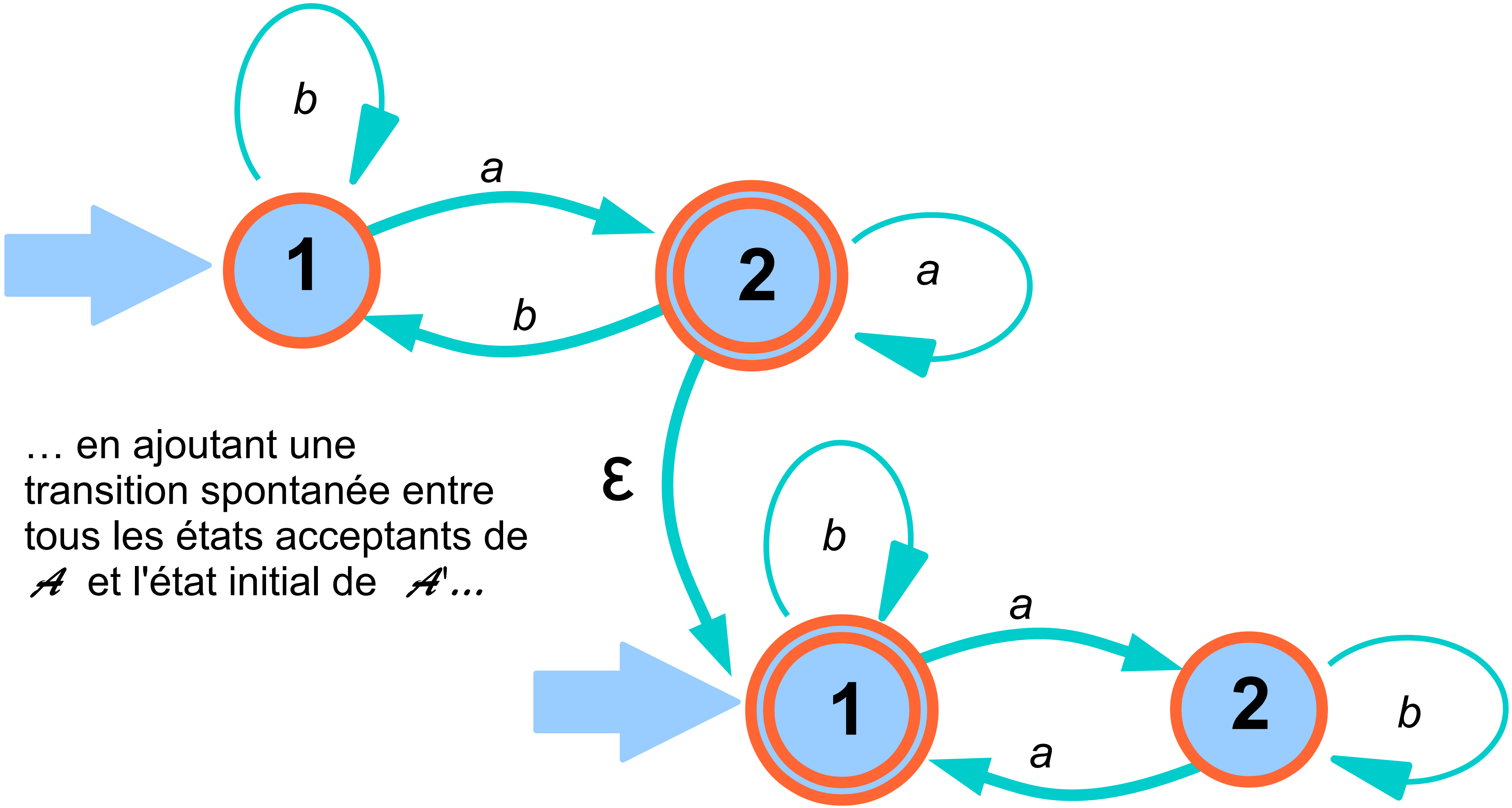
Produit

L'automate produit est donc obtenu en juxtaposant les automates $\mathcal{A}$ et $\mathcal{A}'$ ...





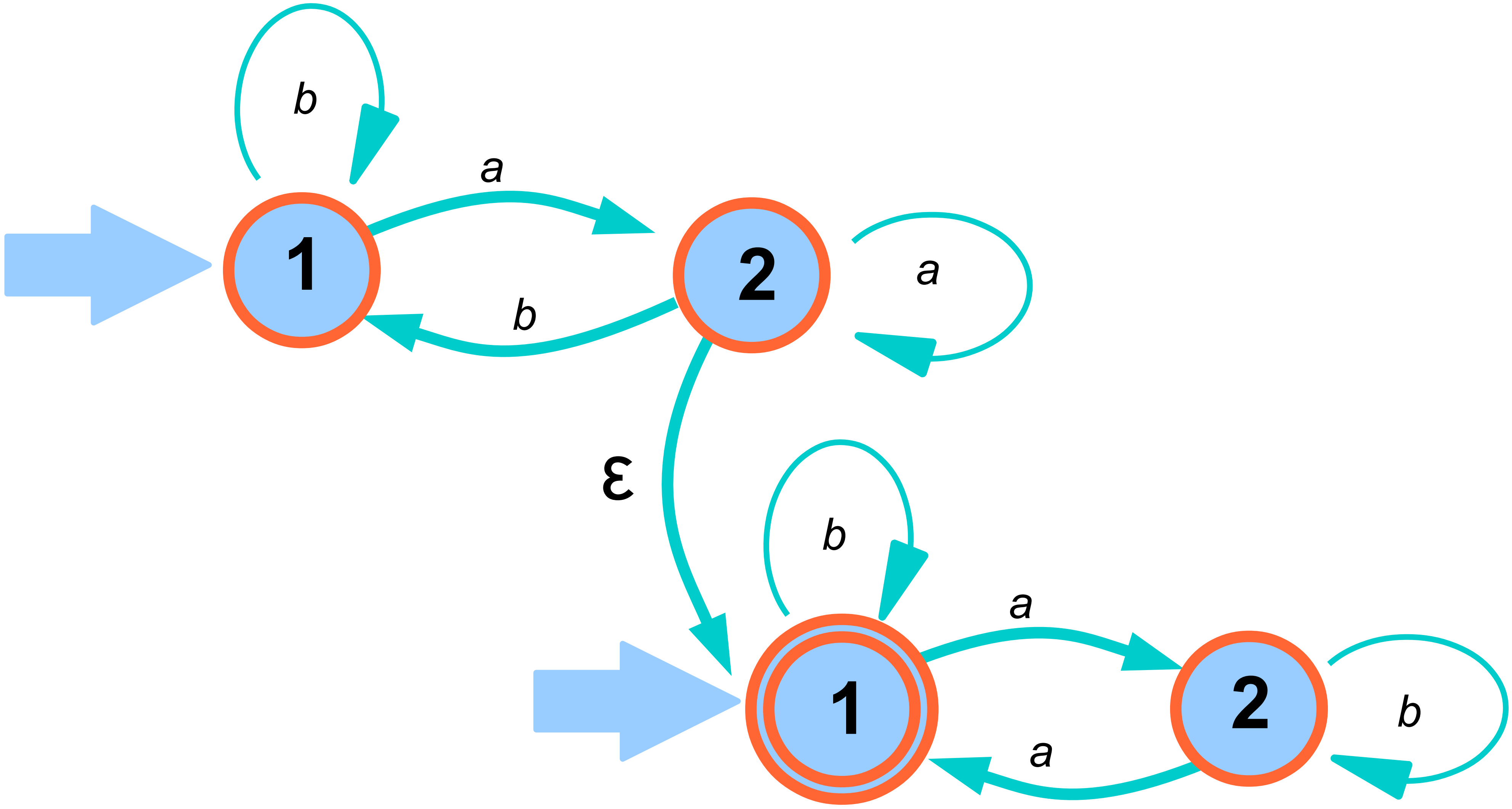
Produit



… en ajoutant une transition spontanée entre tous les états acceptants de $\mathcal{A}$ et l'état initial de $\mathcal{A}'$…

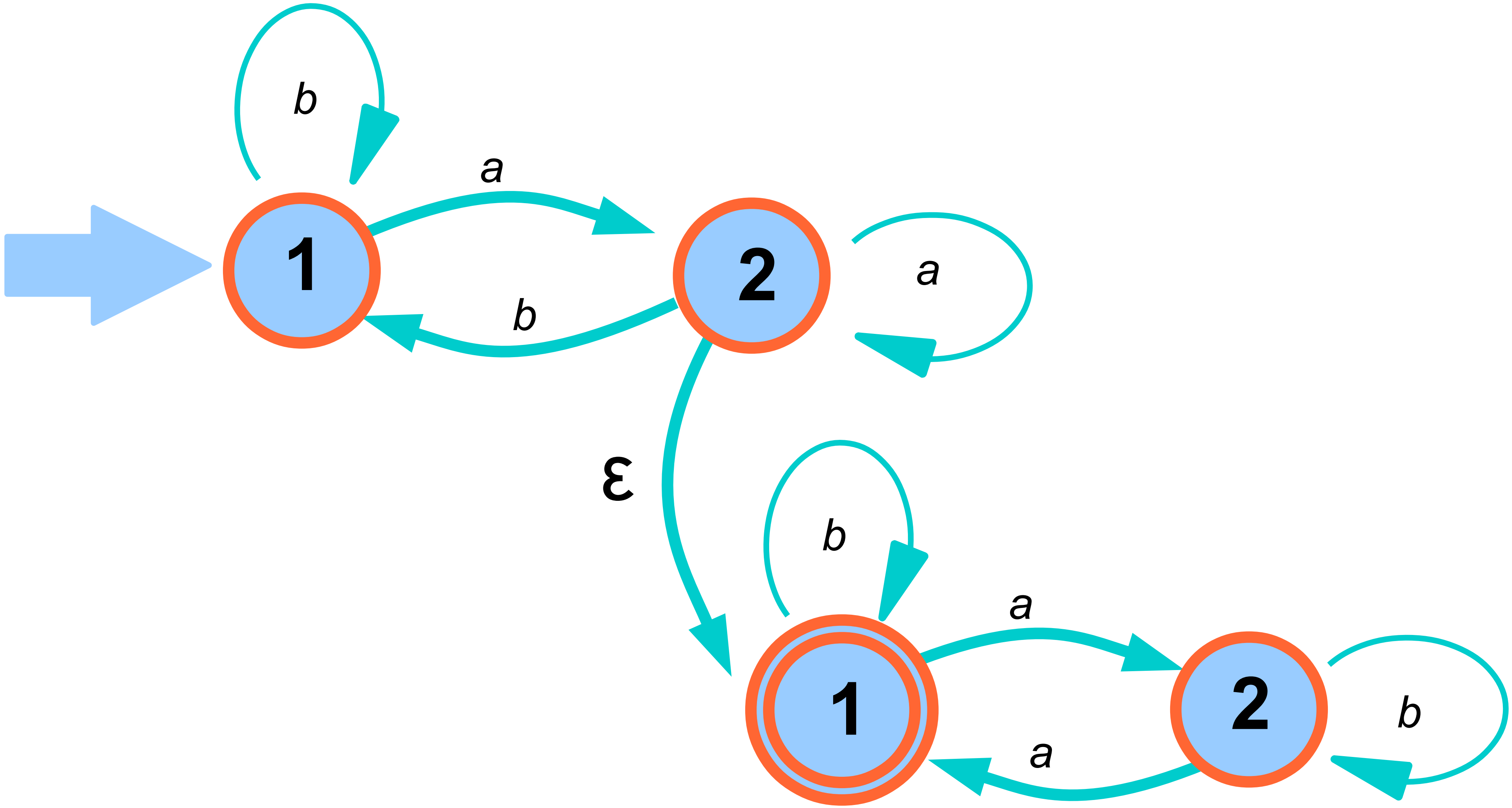
Produit

...en rendant refusant les états acceptant de $\mathcal{A}$ ...



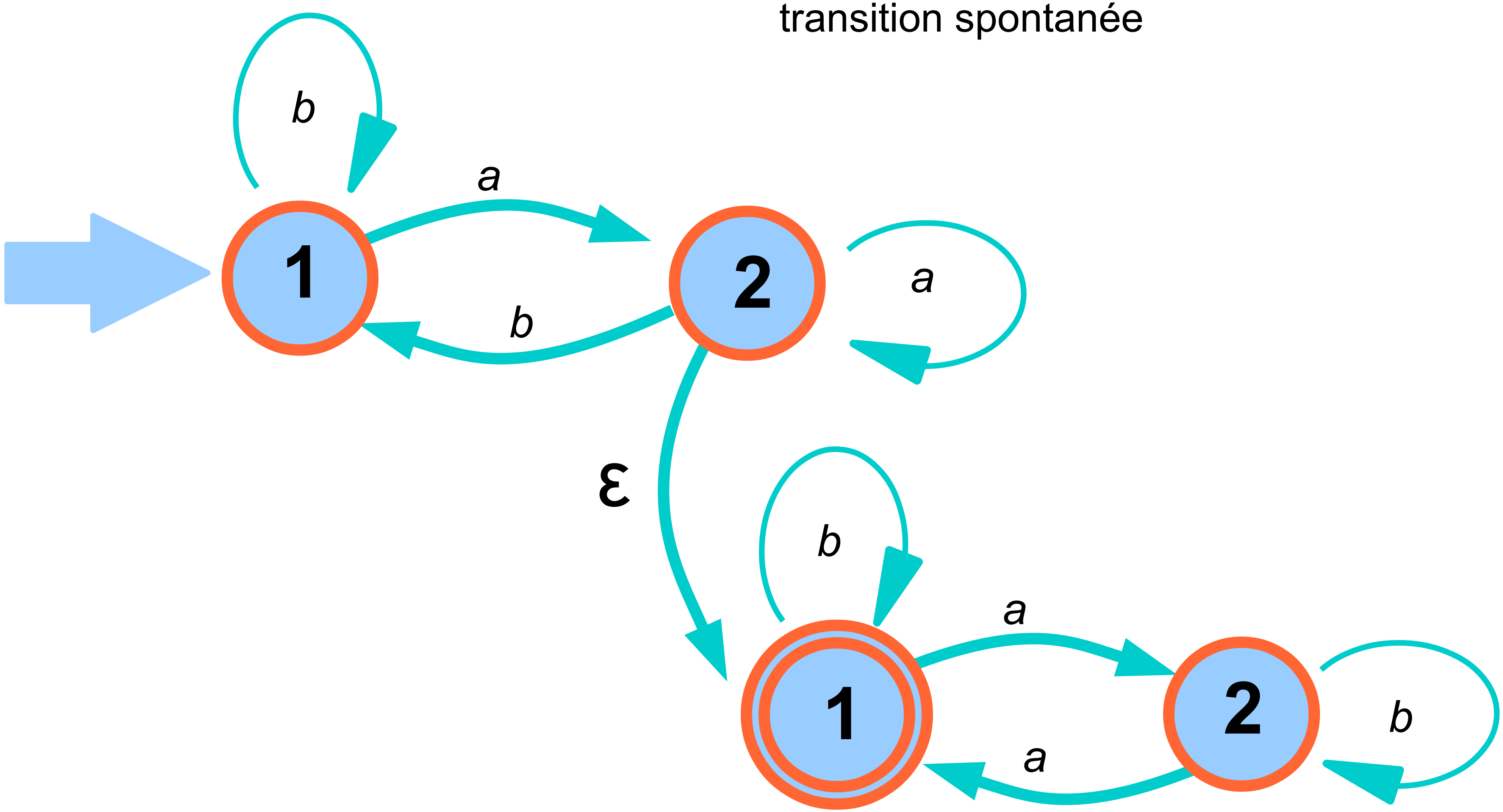
Produit

… et en enlevant le caractère initial à l'état initial de $\mathscr{A}'$.

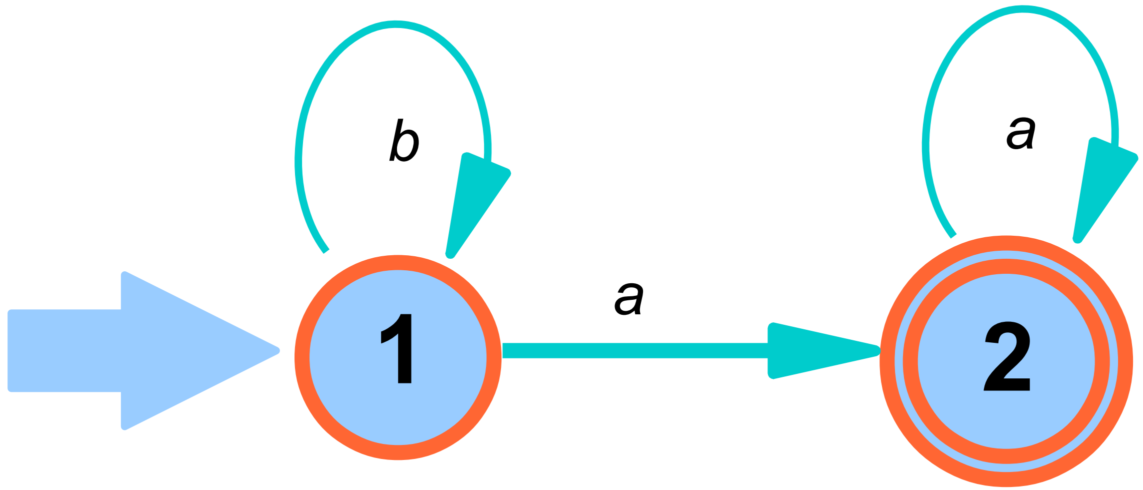


Produit

- On obtient un automate à n+p états
- Non déterministe à cause de la transition spontanée



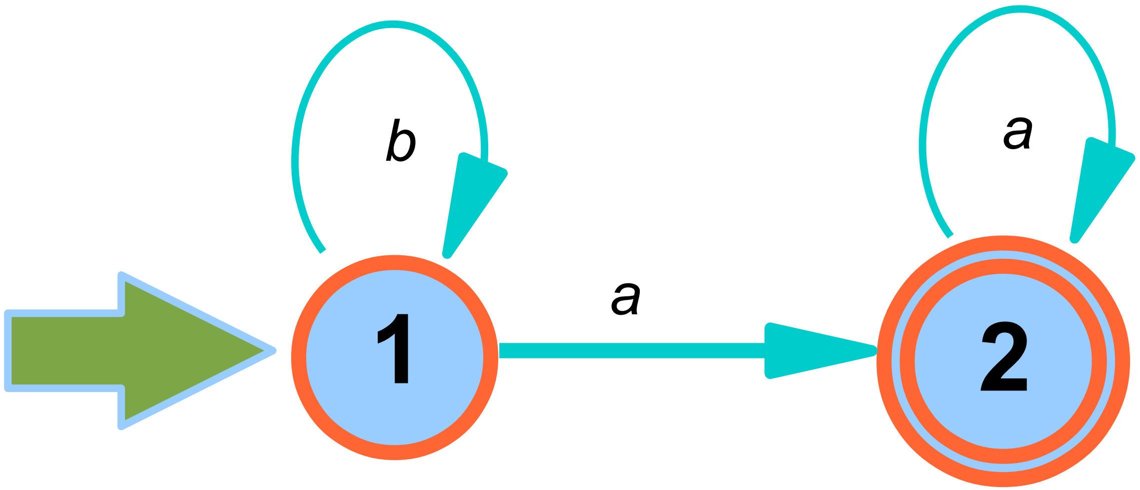
Langage L+



On considère l'automate $\mathcal{A}$
reconnaissant le langage L des
mots contenant au moins un
« a » mais pas « ab »

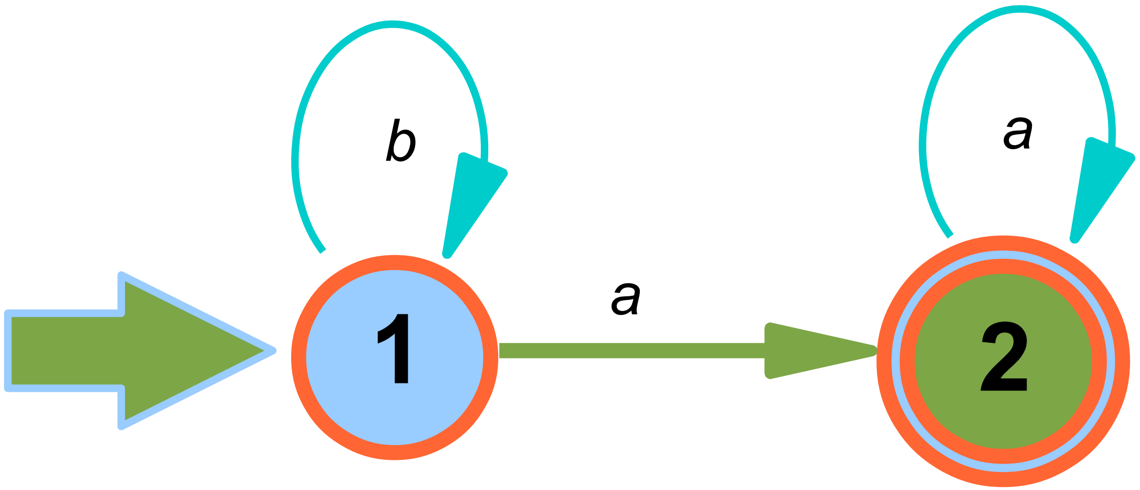
Langage L+

Le mot « abba » = « a.bba » est dans le langage L+. On peut le lire en enchaînant deux lectures par l'automate $\mathscr{A}$



Langage L+

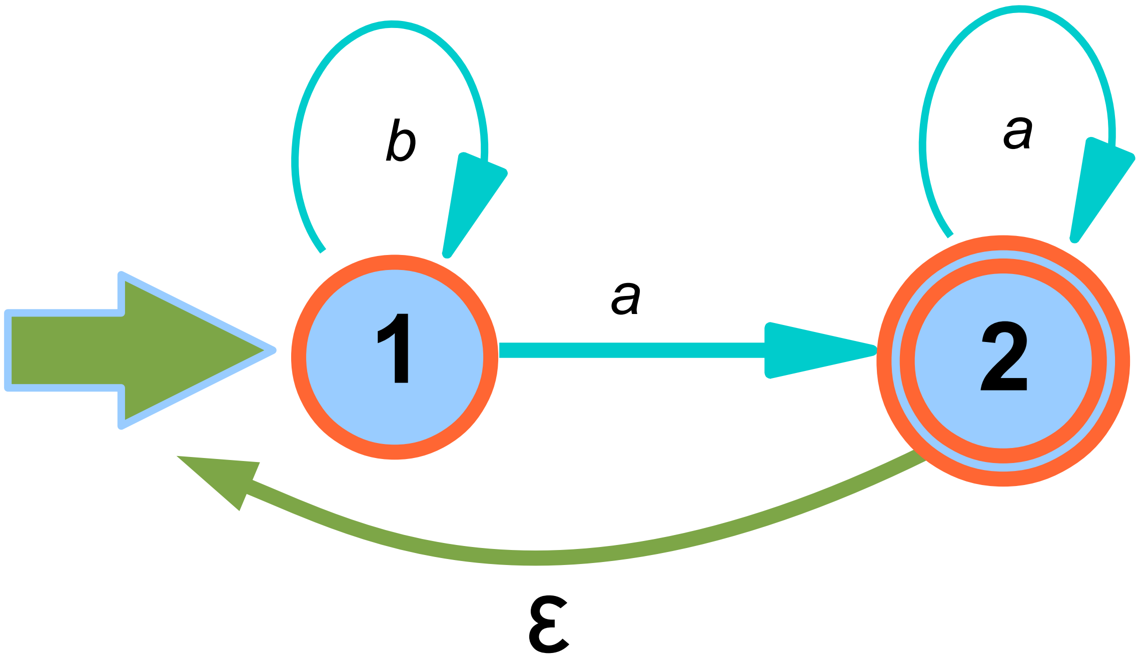
Le mot « abba » = « a.bba » est dans le langage L+. On peut le lire en enchaînant deux lectures par l'automate $\mathcal{A}$



lecture acceptante de « a » par l'automate $\mathcal{A}$

Langage L+

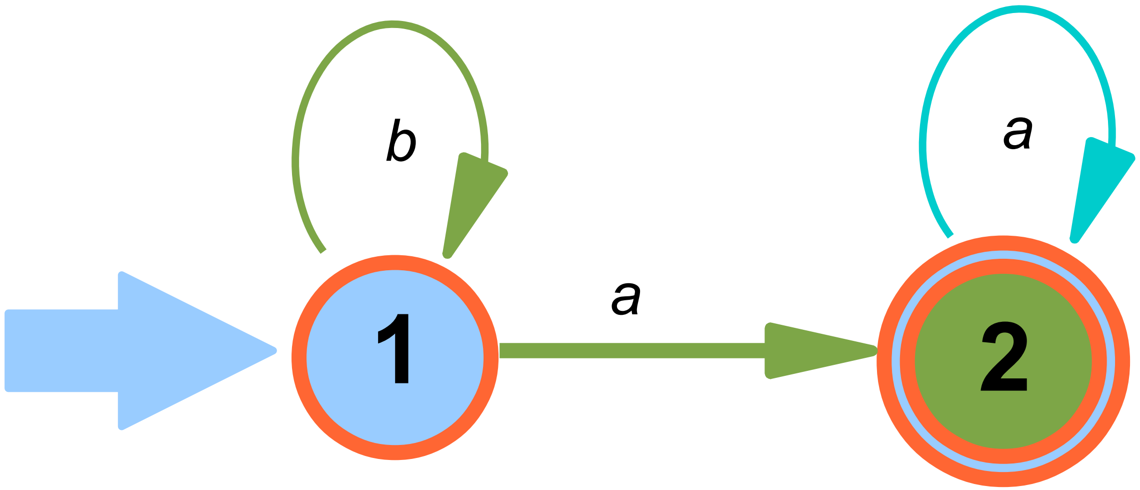
Le mot « abba » = « a.bba » est dans le langage L+. On peut le lire en enchaînant deux lectures par l'automate $\mathcal{A}$



Retour spontané à l'état initial

Langage L+

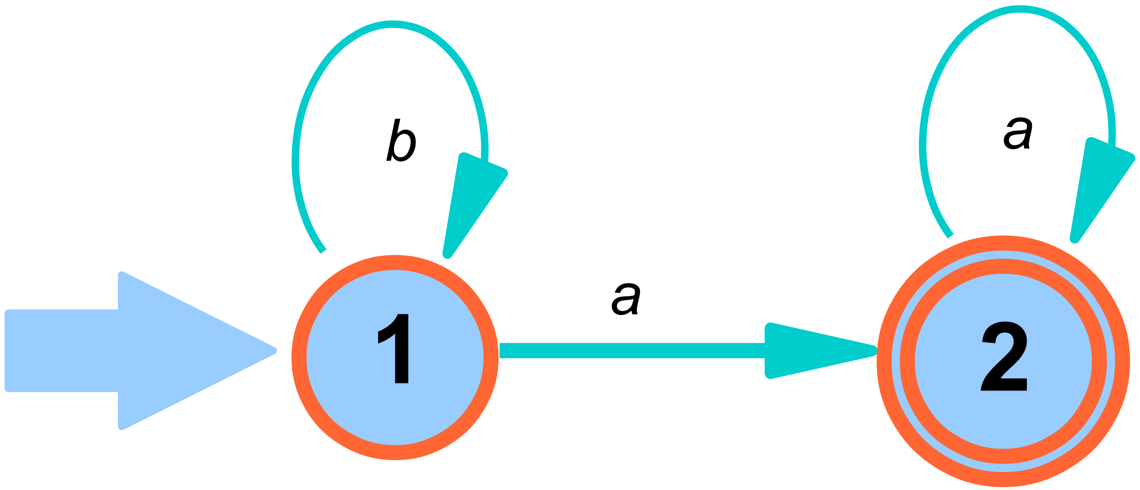
Le mot « abba » = « a.bba » est dans le langage L+. On peut le lire en enchaînant deux lectures par l'automate $\mathcal{A}$



lecture acceptante de « bba » par l'automate $\mathcal{A}$

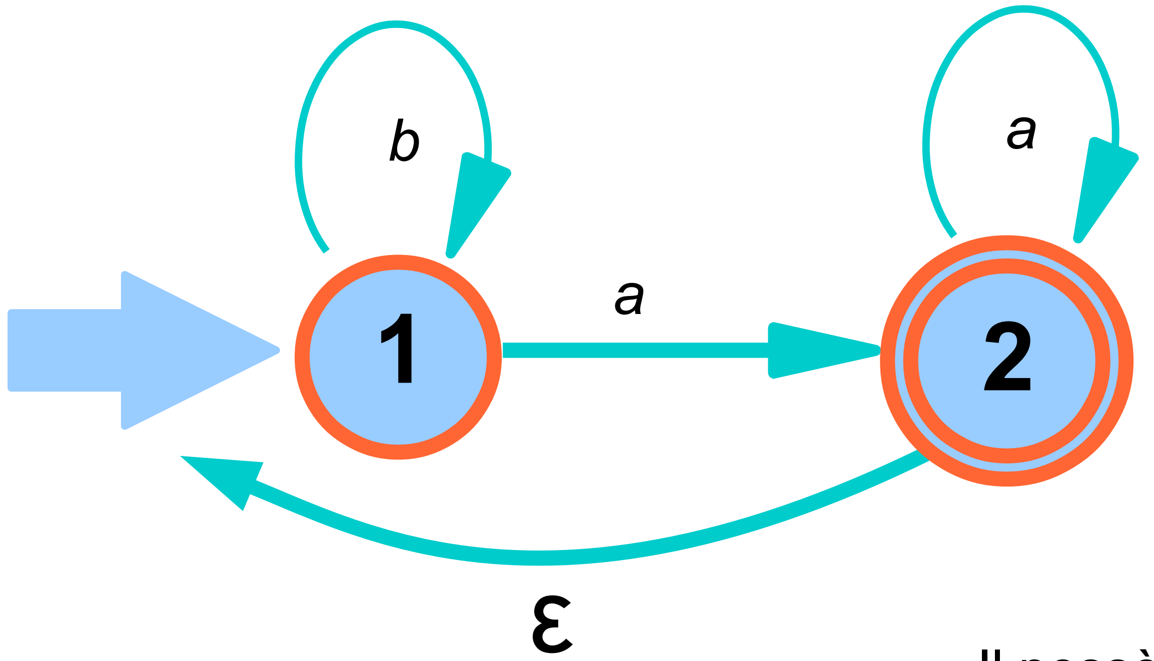
Langage L+

L'automate reconnaissant L+ est donc obtenu à partir de l'automate $\mathcal{A}$ ...



Langage L+

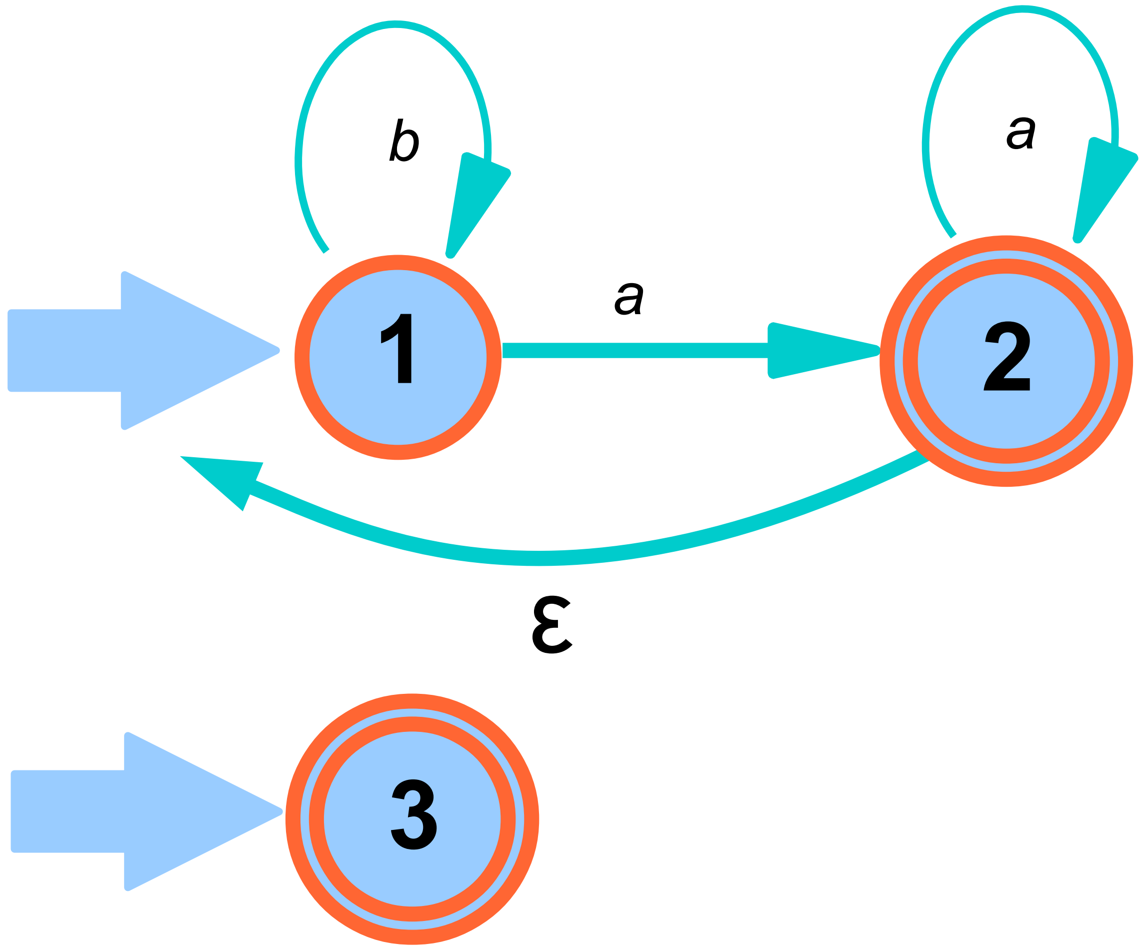
… en ajoutant une transition spontanée de tous les états acceptant vers l'état initial.



• Il possède n états
• Mais n'est pas déterministe du fait de la transition spontanée

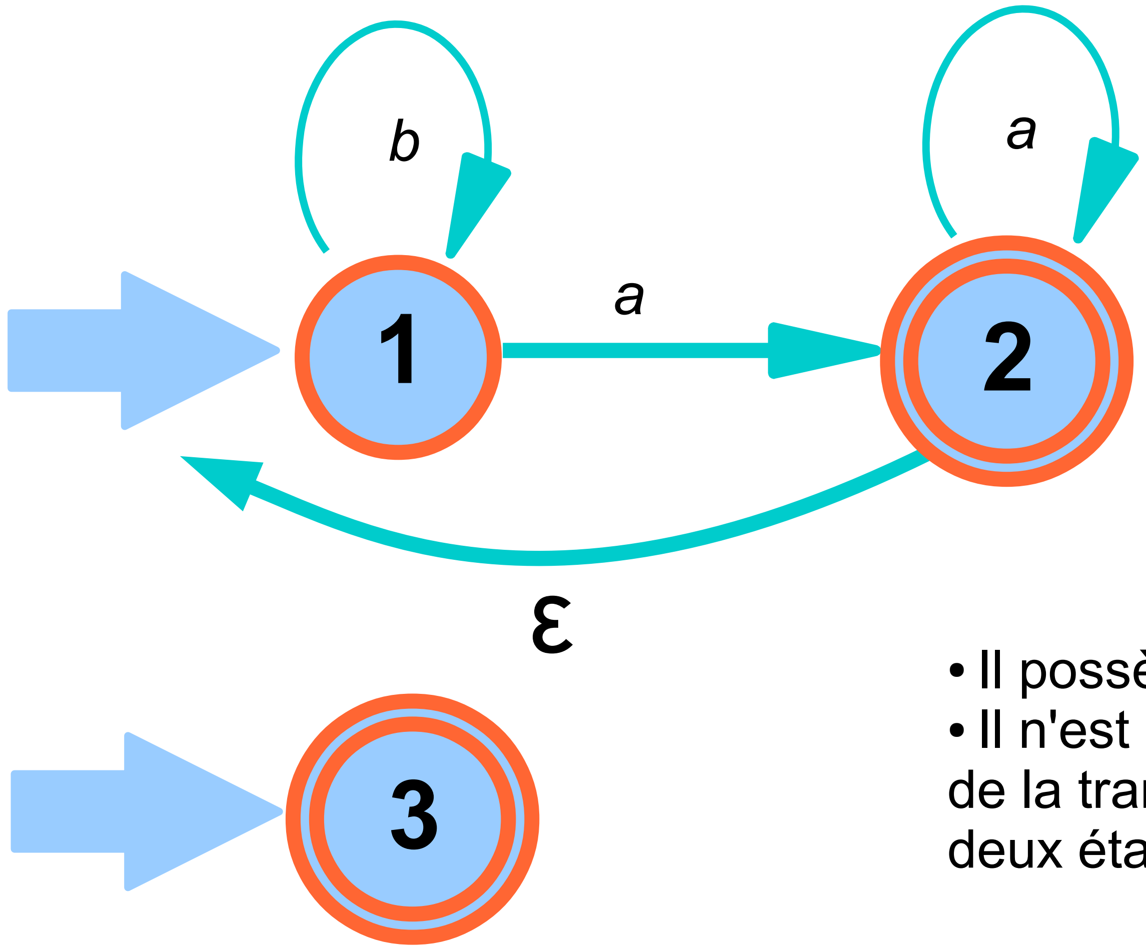
Langage L*

L'automate reconnaissant L* est obtenu à partir du précédent en ajoutant un état initial acceptant pour reconnaître ε.



Langage L*

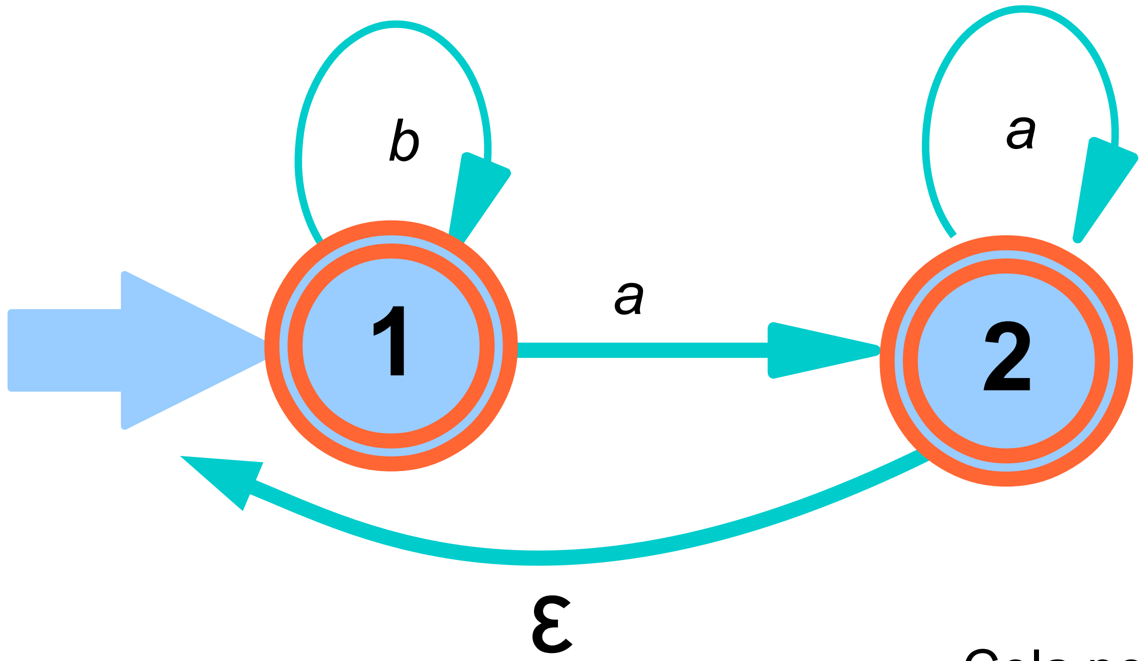
L'automate reconnaissant L* est obtenu à partir du précédent en ajoutant un état initial acceptant pour reconnaître ε.



- Il possède n+1 états
- Il n'est pas déterministe du fait de la transition spontanée et des deux états initiaux

Langage L*

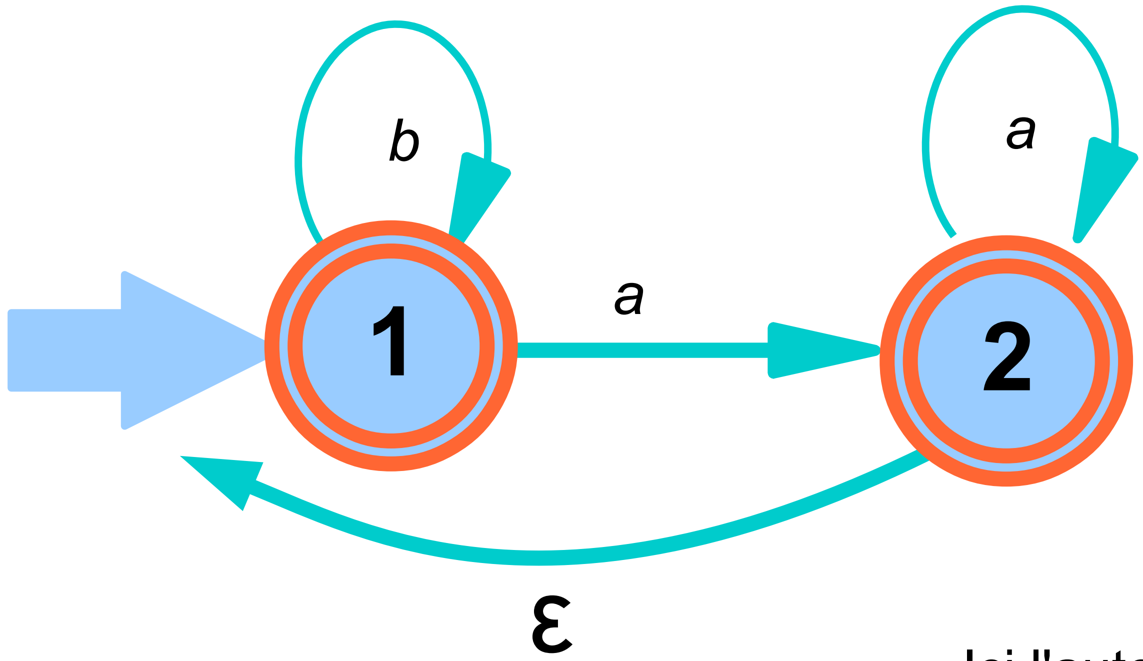
ATTENTION !
Vous êtes parfois tentés de rendre l'état initial acceptant plutôt que d'ajouter un nouvel état initial acceptant.



Cela permet bien de reconnaître ε, mais aussi d'autre mots non voulus

Langage L*

ATTENTION !
Vous êtes parfois tentés de rendre l'état initial acceptant plutôt que d'ajouter un nouvel état initial acceptant.



Ici l'automate reconnaît « bb » qui n'appartient pas à L*