Extended Branching Rauzy Induction and IET Return Word Clustering

Christian B. Hughes

Czech Technical University in Prague

Prague, Czech Republic 2026

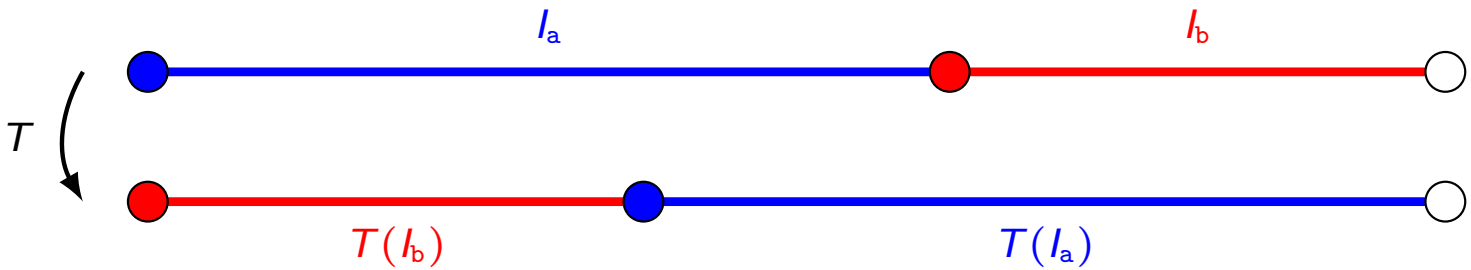
Branching Rauzy Induction

(A Quick Recap)



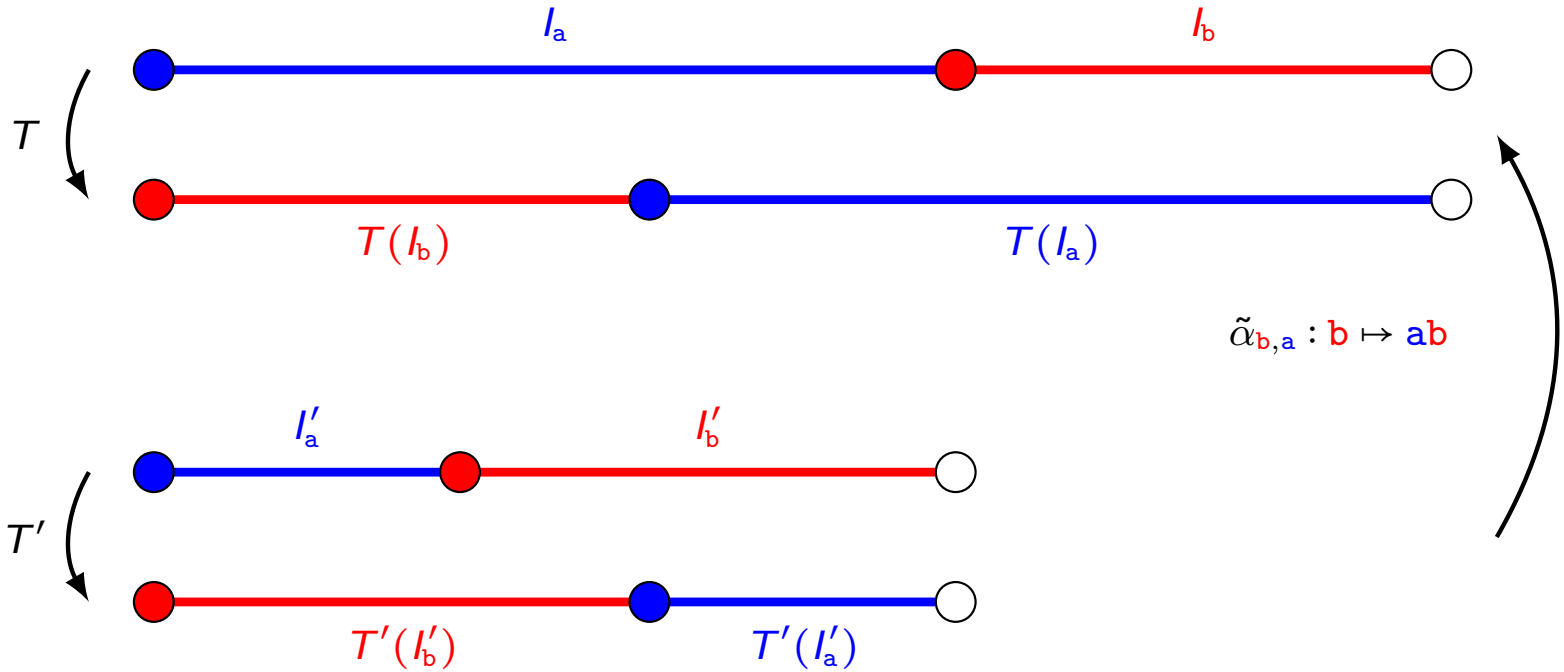
Branching Rauzy Induction

(A Quick Recap)



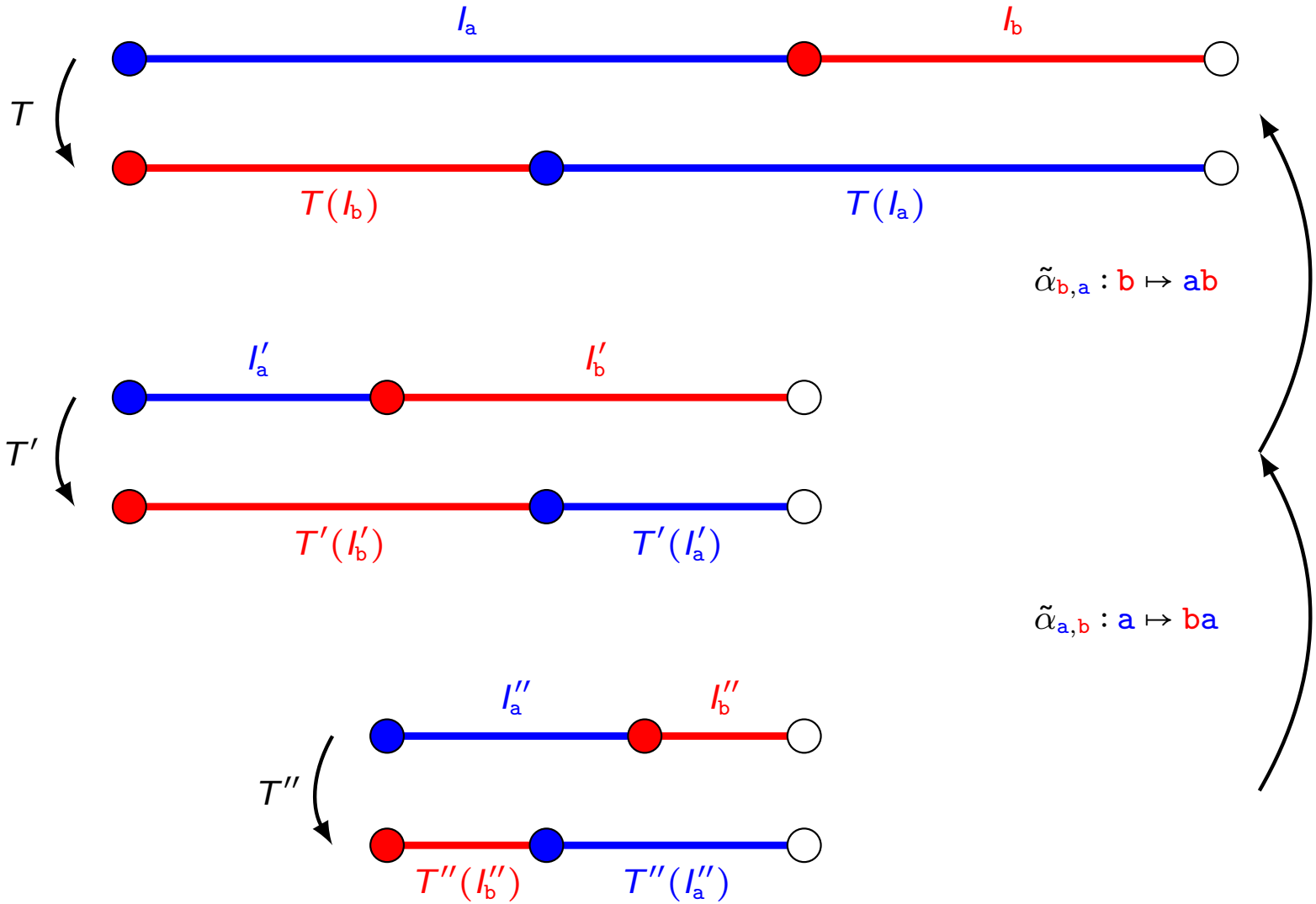
Branching Rauzy Induction

(A Quick Recap)



Branching Rauzy Induction

(A Quick Recap)



Branching Rauzy Induction and Return Words



Branching Rauzy Induction and Return Words

Proposition [Dolce, Perrin 2018]

Let T be a regular IET over the alphabet $\mathcal{A}$. Let $w \in \mathcal{L}(T)$. There exists a finite sequence χ comprised of right and left induction steps such that $\chi(T)$ is the induced transformation of T on the cylinder I_w.

Branching Rauzy Induction and Return Words

Proposition [Dolce, Perrin 2018]

Let T be a regular IET over the alphabet $\mathcal{A}$. Let $w \in \mathcal{L}(T)$. There exists a finite sequence χ comprised of right and left induction steps such that $\chi(T)$ is the induced transformation of T on the cylinder I_w.

Theorem [Dolce, Perrin 2018]

The set of return words to w is precisely the set $\{\theta(a) \mid a \in \mathcal{A}\}$, where θ is the composition of the induction morphisms associated to χ from the "end" of the induction to the "beginning."

And if the IET is Not Regular?



CTU
CZECH TECHNICAL UNIVERSITY IN PRAGUE

And if the IET is Not Regular?

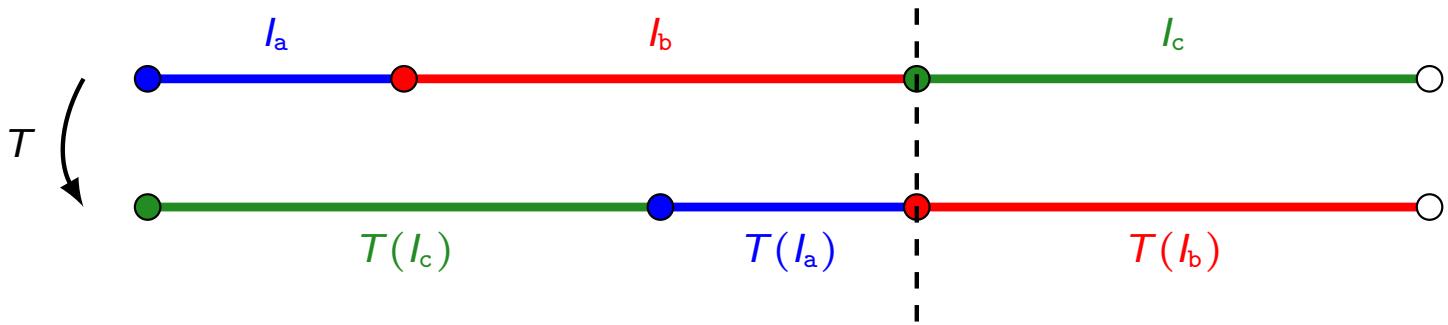
We cannot induce on I_a via standard branching Rauzy induction.



CTU
CZECH TECHNICAL UNIVERSITY IN PRAGUE

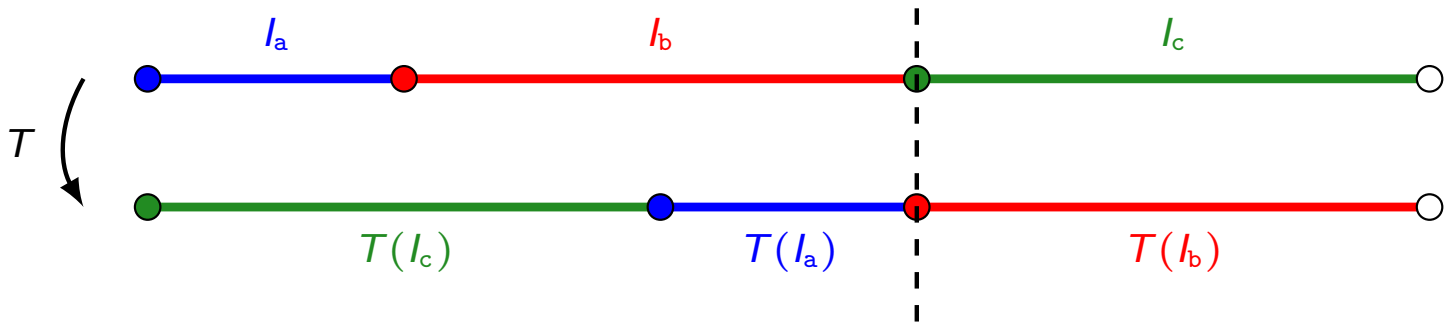
And if the IET is Not Regular ?

We cannot induce on I_a via standard branching Rauzy induction.



And if the IET is Not Regular?

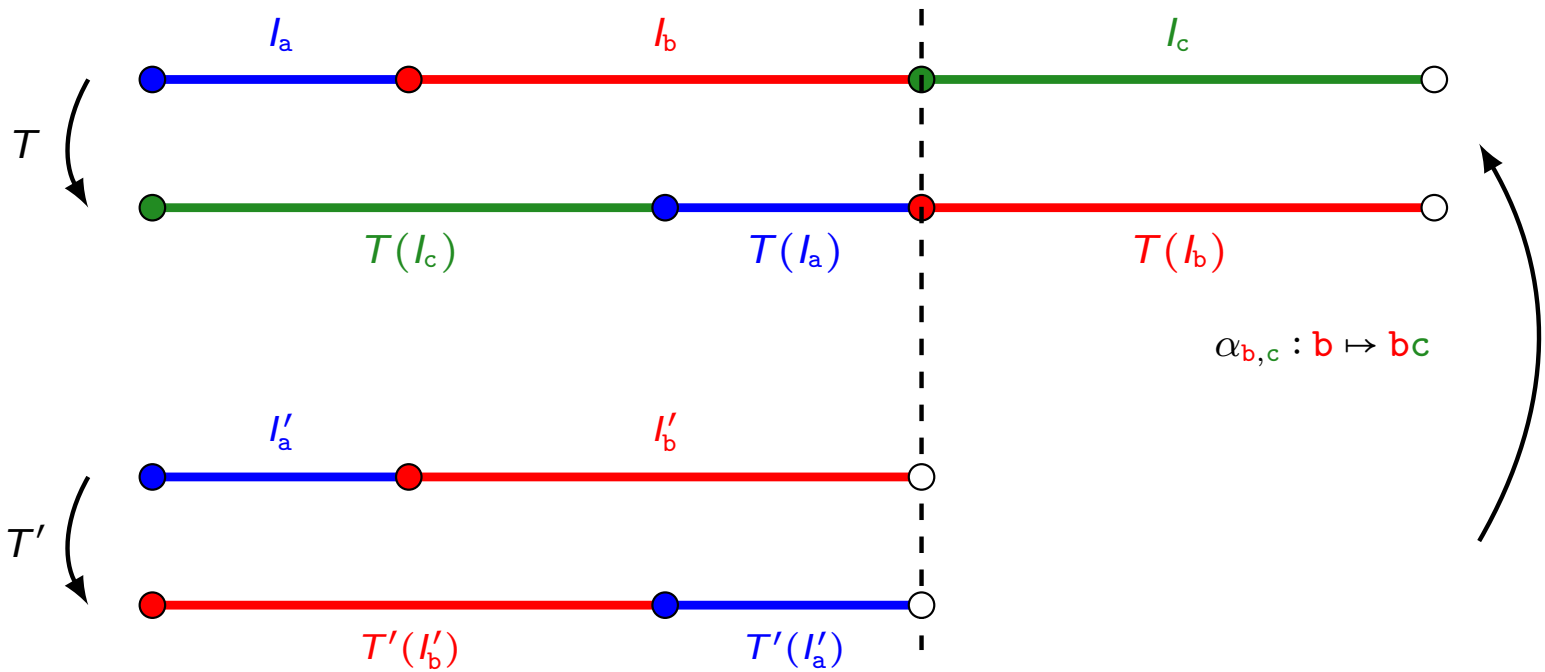
We cannot induce on I_a via standard branching Rauzy induction.



But notice that every b is immediately followed by a c in $\mathcal{L}(T)$.

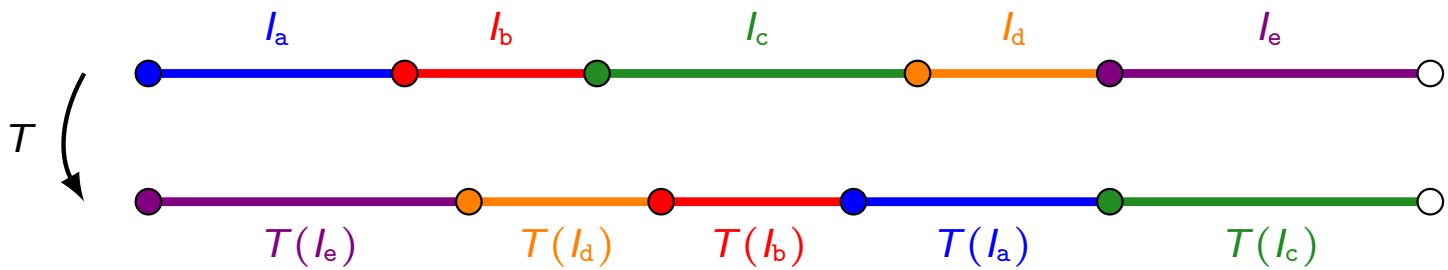
Merging

Merging

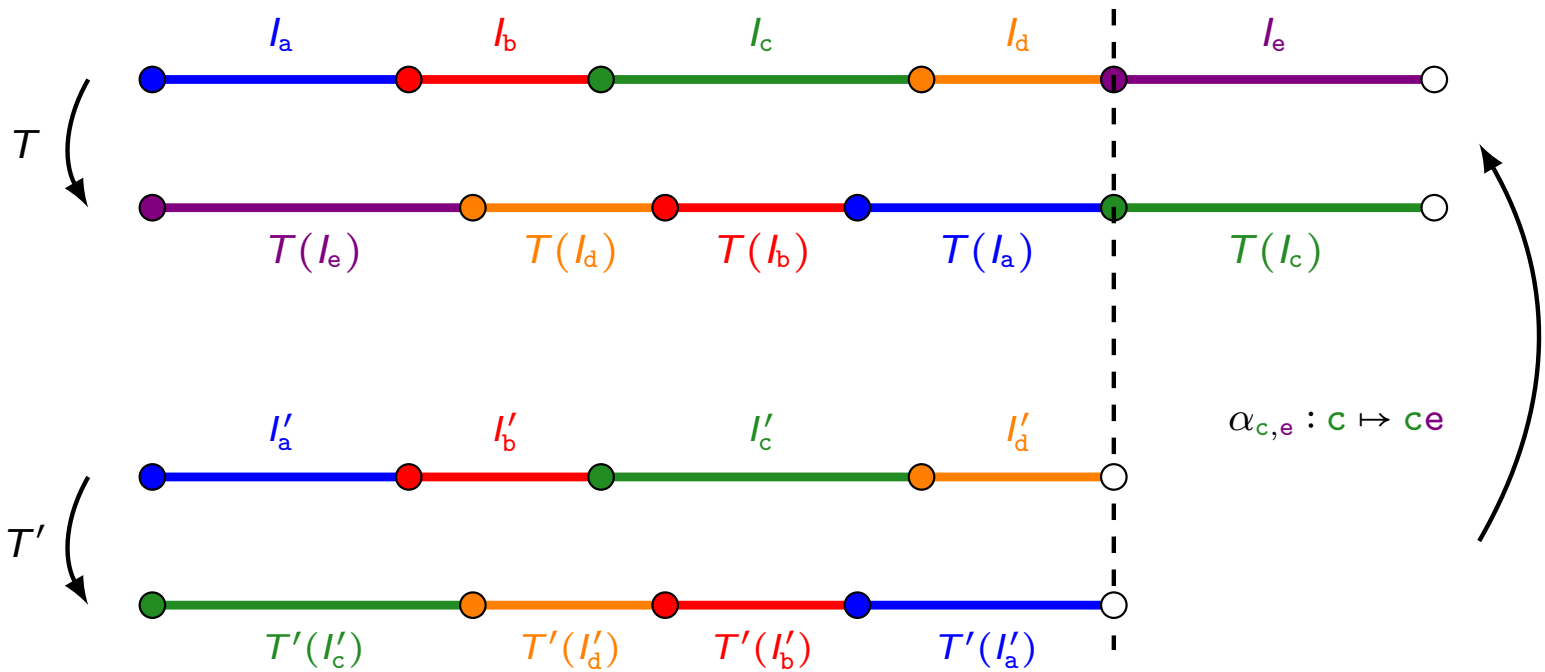


Another Example

Another Example



Another Example

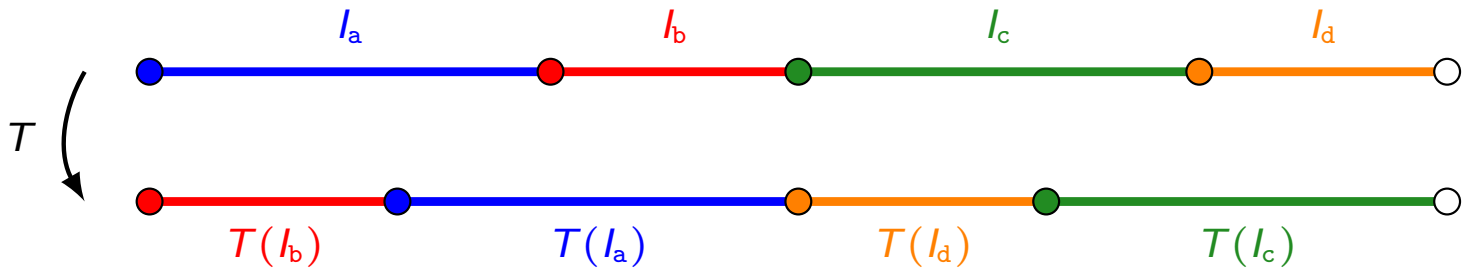


Any Other Obstructions ?



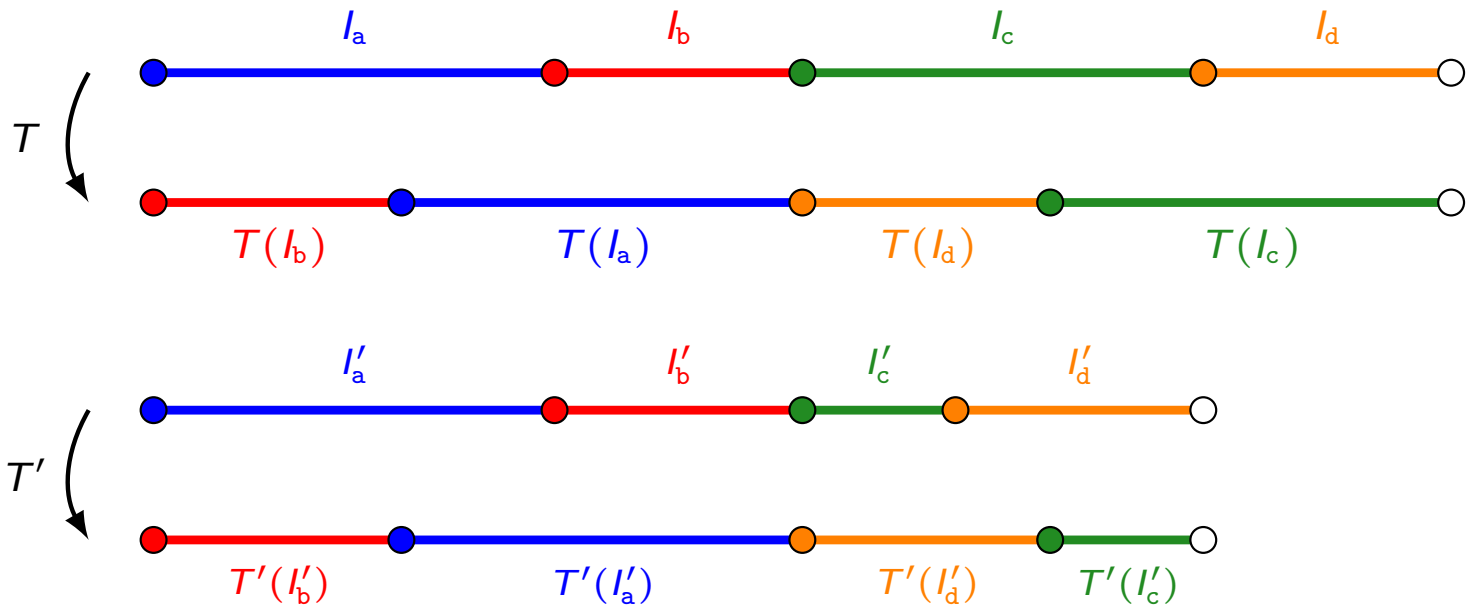
Any Other Obstructions ?

What if we try to induce on I_a ?



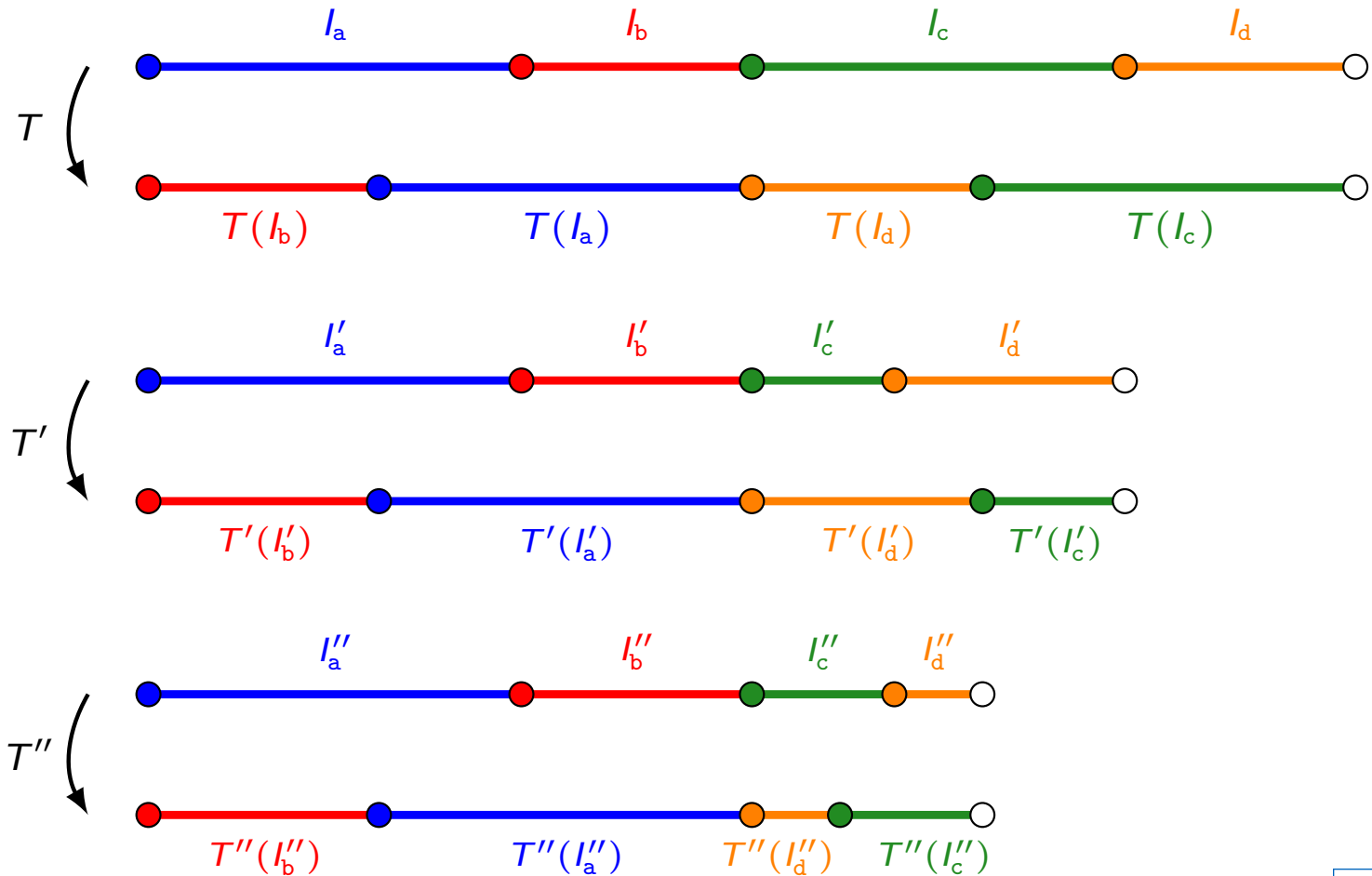
Any Other Obstructions ?

What if we try to induce on I_a ?

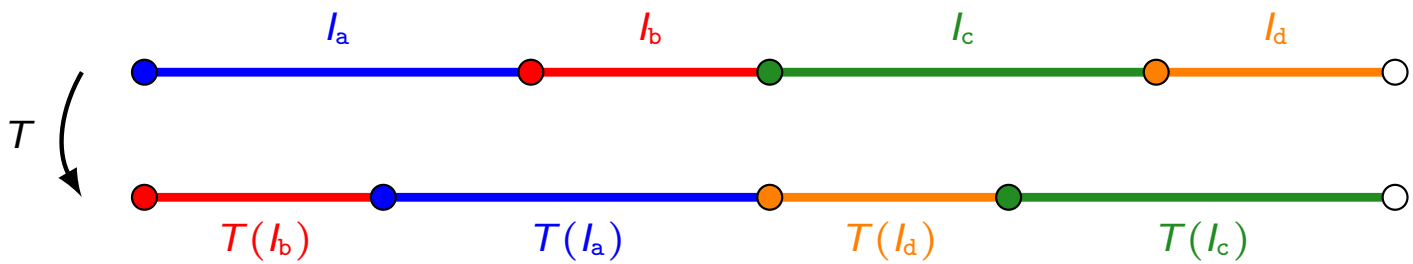


Any Other Obstructions ?

What if we try to induce on I_a ?

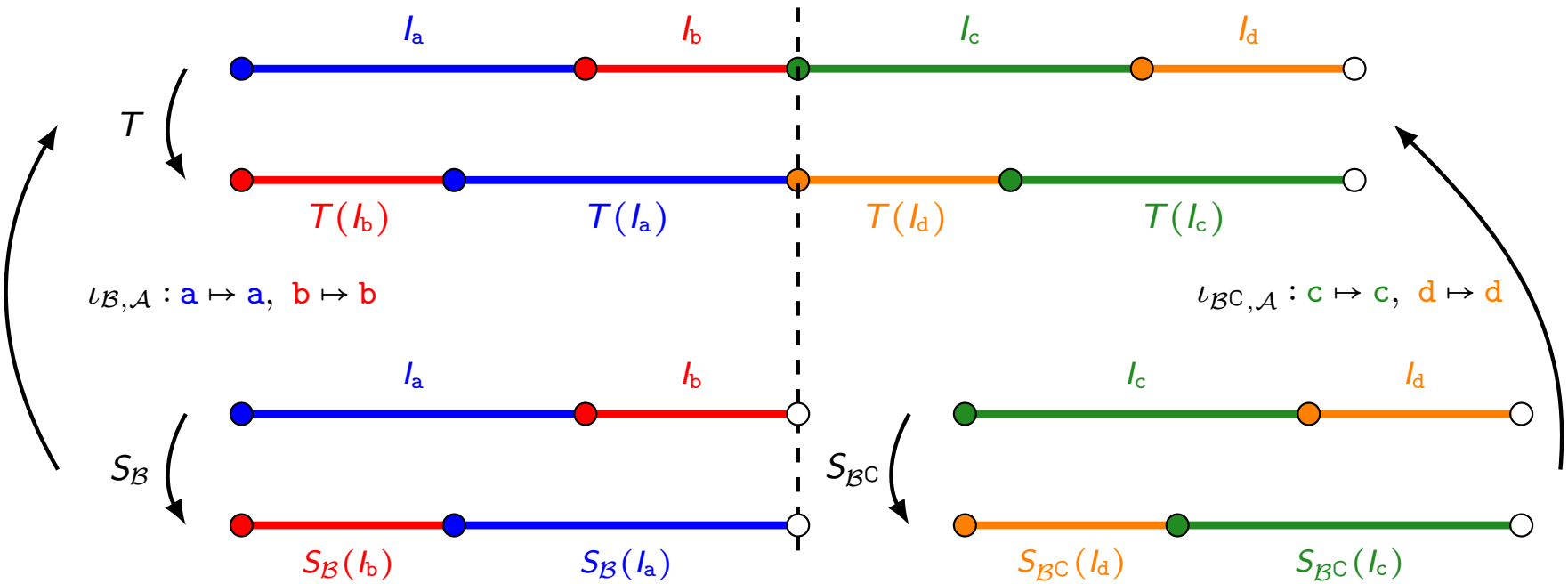


Splitting



Splitting

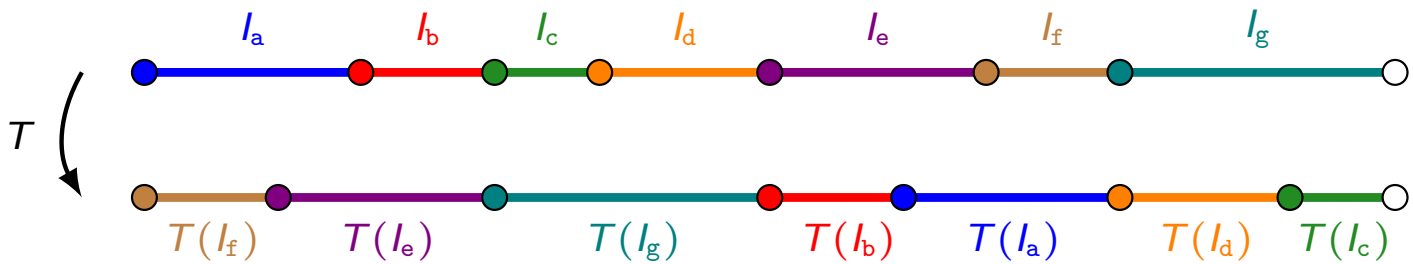
We can *split* T over the alphabet $\mathcal{A} = \{\mathtt{a} < \mathtt{b} < \mathtt{c} < \mathtt{d}\}$ into two IETs $S_{\mathcal{B}}$ and $S_{\mathcal{B}^{\mathsf{C}}}$ over the alphabets $\mathcal{B} = \{\mathtt{a} < \mathtt{b}\}$ and $\mathcal{B}^{\mathsf{C}} = \{\mathtt{c} < \mathtt{d}\}$ respectively.



Splitting Domains Need Not Be Contiguous

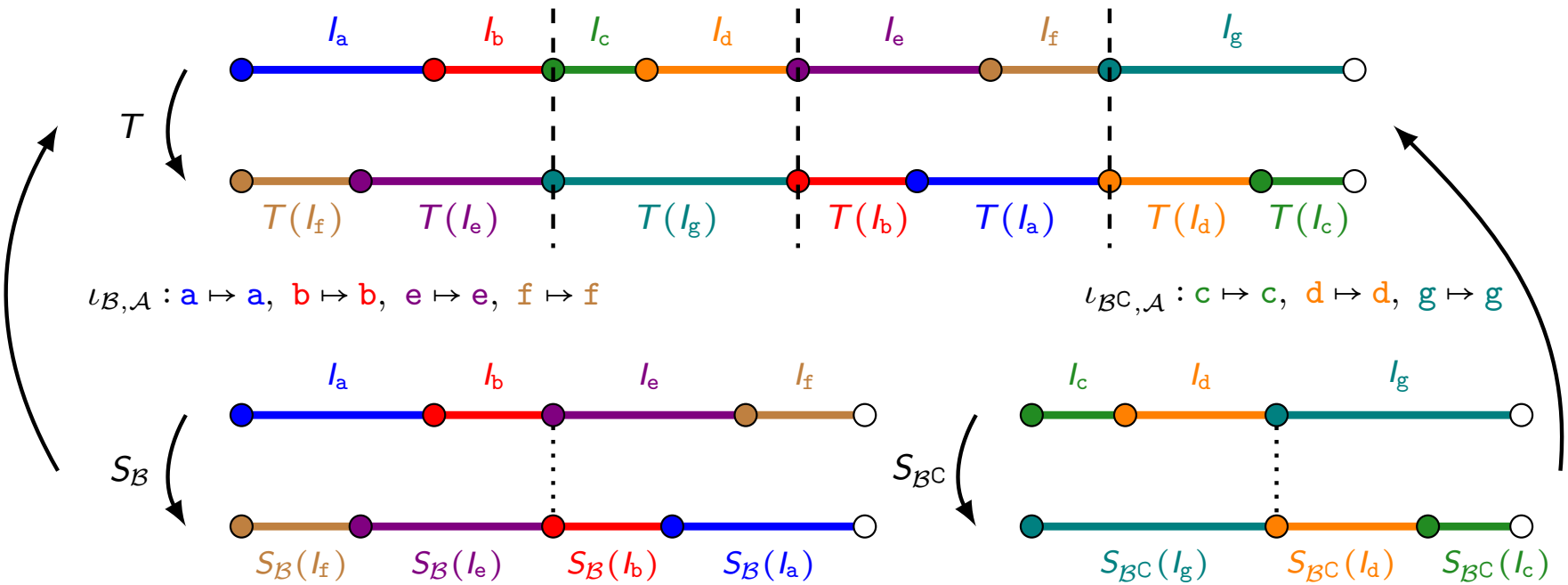


Splitting Domains Need Not Be Contiguous



Splitting Domains Need Not Be Contiguous

The intervals $I_a \cup I_b$ and $I_e \cup I_f$ are glued together to form the domain of $S_{\mathcal{B}}$



Induction on Cylinders for Arbitrary Standard IETs



CTU
CZECH TECHNICAL UNIVERSITY IN PRAGUE

Induction on Cylinders for Arbitrary Standard IETs

Proposition [Dolce, H. 2025+]

Let T be a standard IET over the alphabet $\mathcal{A}$. Let $w \in \mathcal{L}(T)$. There exists a finite sequence χ comprised of splitting, merging, right, and left induction steps such that $\chi(T)$ is the induced transformation of T on the cylinder I_w.



CTU
CZECH TECHNICAL UNIVERSITY IN PRAGUE

Induction on Cylinders for Arbitrary Standard IETs

Proposition [Dolce, H. 2025+]

Let T be a standard IET over the alphabet $\mathcal{A}$. Let $w \in \mathcal{L}(T)$. There exists a finite sequence χ comprised of splitting, merging, right, and left induction steps such that $\chi(T)$ is the induced transformation of T on the cylinder I_w.

Theorem [Dolce, H. 2025+]

The set of return words to w is precisely the set $\{\theta(a) \mid a \in \mathcal{A}'\}$, where θ is the composition of the induction morphisms associated to χ from the "end" of the induction to the "beginning," and $\mathcal{A}'$ is the alphabet of the IET induced on I_w.

Cyclic Conjugate Intermezzo



Cyclic Conjugate Intermezzo

Let w = *vezmeme* over the ordered Czech alphabet $\check{\mathcal{C}} = \{a < \acute{a} < \ldots < \check{z}\}$.



Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

zmemeve

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

zmemeve

memevez

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

zmemeve

memevez

emevezm

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

zmemeve

memevez

emevezm

mevezme

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

vezmeme

ezmemev

zmemeve

memevez

emevezm

mevezme

evezmem

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

emevezm

evezmem

ezmemev

memevez

mevezme

vezmeme

zmemeve

Cyclic Conjugate Intermezzo

Let $w = vezmeme$ over the ordered Czech alphabet $\check{C} = \{a < á < \ldots < ž\}$.

$$\begin{bmatrix} \texttt{emevezm} \\ \texttt{evezmem} \\ \texttt{ezmemev} \\ \texttt{memevez} \\ \texttt{mevezme} \\ \texttt{vezmeme} \\ \texttt{zmemeve} \end{bmatrix}$$

The Burrows-Wheeler Transform

$$\begin{bmatrix} \texttt{emevezm} \\ \texttt{evezmem} \\ \texttt{ezmemev} \\ \texttt{memevez} \\ \texttt{mevezme} \\ \texttt{vezmeme} \\ \texttt{zmemeve} \end{bmatrix}$$

The Burrows-Wheeler Transform

$$\begin{bmatrix} \texttt{emevezm} \\ \texttt{evezmem} \\ \texttt{ezmemev} \\ \texttt{memevez} \\ \texttt{mevezme} \\ \texttt{vezmeme} \\ \texttt{zmemeve} \end{bmatrix}$$

The Burrows-Wheeler Transform

$$\begin{bmatrix} \texttt{emevezm} \\ \texttt{evezmem} \\ \texttt{ezmemev} \\ \texttt{memevez} \\ \texttt{mevezme} \\ \texttt{vezmeme} \\ \texttt{zmemeve} \end{bmatrix}$$

$$\mathrm{bwt}_{\mathcal{C}}(\texttt{vezmeme}) = \texttt{mmvzeee}$$

Clustering Words



Clustering Words

A word w is *π-clustering* if $\mathrm{bwt}_{\mathcal{A}}(w) = a_{\pi(a_1)}^{k_1} a_{\pi(a_2)}^{k_2} \cdots a_{\pi(a_n)}^{k_n}$. The word $w =$ vezmeme clusters with $\pi = (1\,2\,3\,4)$ when considered as a word over $\mathcal{A} = \{e < m < v < z\}$.

Clustering of Return Words in Languages of IETs



Clustering of Return Words in Languages of IETs

Proposition [Dolce, H. 2025+]

Let T be a standard IET over the alphabet $\mathcal{A}$ with permutation π and let $w \in \mathcal{L}(T)$. If $u \in \mathcal{L}(T)$ is a return word to w, then u is π-clustering.

Děkuji za pozornost !

