Topological properties of surfaces flows

Tomoo YOKOYAMA

Department of Mathematics, Kyoto University of Education / JST

13th March 2017
@London

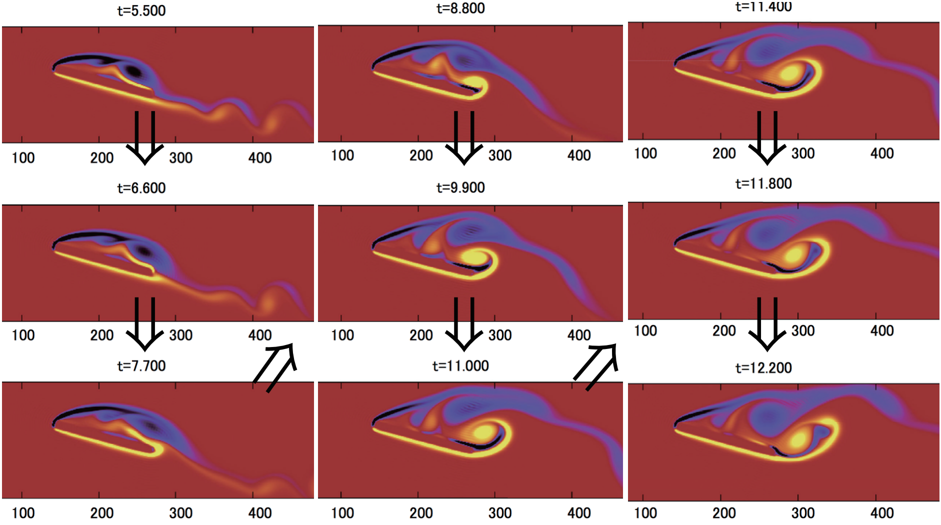
Contents

1. Applications and Background
2. Rep of uniform flows with pt vortices
3. Rep of surface flows
4. Topological str and Data str of surface flows

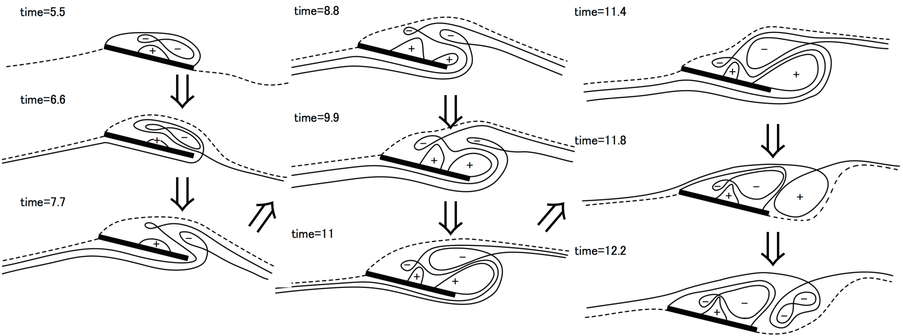
1. Applications and Background
2. Rep of uniform flows with pt vortices
3. Rep of surface flows
4. Topological str and Data str of surface flows

A problem and a topological approch

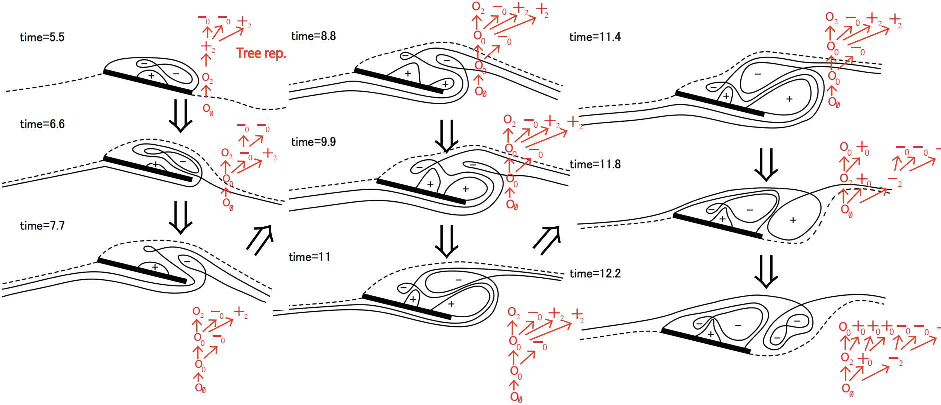
Find extremal lift-to-drag ratios



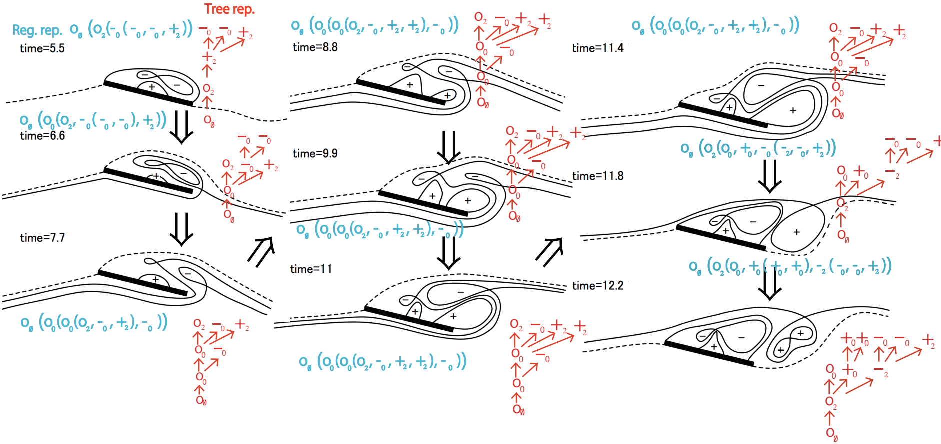
Find extremal lift-to-drag ratios



Find extremal lift-to-drag ratios



Find extremal lift-to-drag ratios



Find extremal lift-to-drag ratios

time = 5.5

$o_{\emptyset}\,(o_2(-_0\,(-_0\,, -_0\,, +_2\,))$

time = 6.5 $\Downarrow$

$o_{\emptyset}\,(o_0(o_2, -_0(-_0\,, -_0), +_2))$

time = 7.5 $\Downarrow$

$o_{\emptyset}\,(o_0(o_0(o_2, -_0\,, +_2), -_0\,))$

$\Nearrow$

time = 8.8

$o_{\emptyset}\,(o_0(o_0(o_2, -_0\,, +_2, +_2), -_0\,))$

$\|$

time = 9.9

$o_{\emptyset}\,(o_0(o_0(o_2, -_0\,, +_2, +_2), -_0\,))$

$\|$

time = 11

$o_{\emptyset}\,(o_0(o_0(o_2, -_0\,, +_2, +_2), -_0\,))$

$//$

time = 11.4

$o_{\emptyset}(o_0(o_0(o_2, -_0\,, +_2, +_2), -_0\,))$

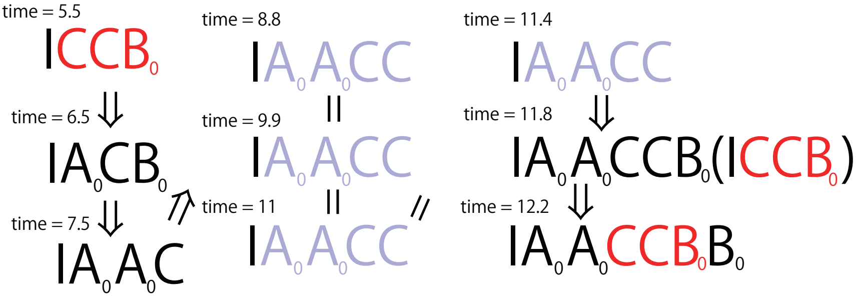
time = 11.8 $\Downarrow$

$o_{\emptyset}(o_2(o_0, +_0, -_0\,(-_2, -_0, +_2))$

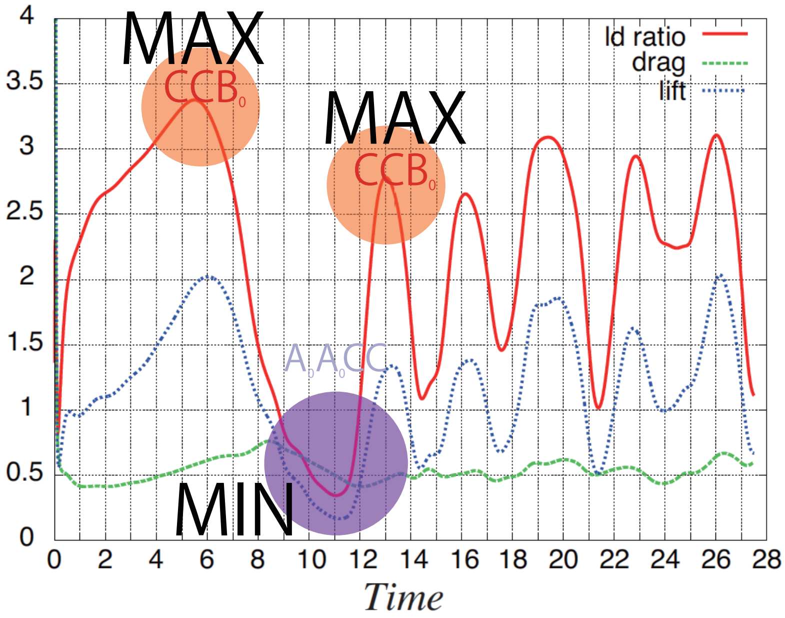
time = 12.2 $\Downarrow$

$o_{\emptyset}(o_2(o_0, +_0(+_0, +_0), -_2\,(-_0, -_0, +_2)))$

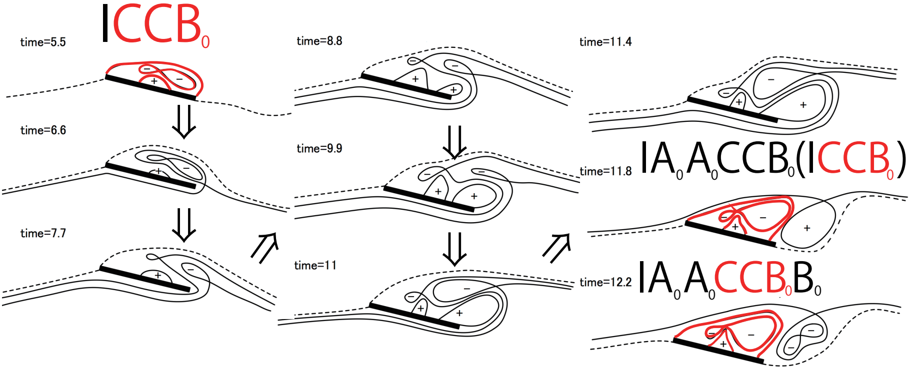
Find extremal lift-to-drag ratios



Evolutions of the lift, the drag and the lift-to-drag ratio



Find high lift-to-drag ratios



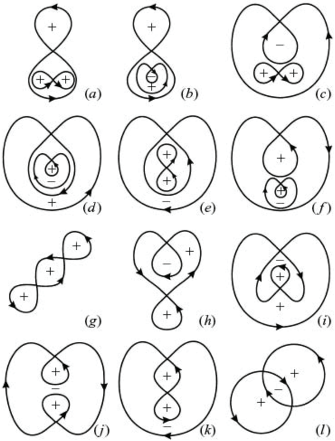
Theoritical background

Topological classification of streamline patterns for Hamiltonian surface flows

◦ Aref and Brøns (1998)
Case: planes without boundaries

◦ Kidambi and Newton (2000)
Case: spheres without boundaries

◦ Moffatt H.K. (2001)
Case: tori without boundaries



Aref & Brøns (JFM:1998)

1. Applications and Background

2. Rep of uniform flows with pt vortices

3. Rep of surface flows

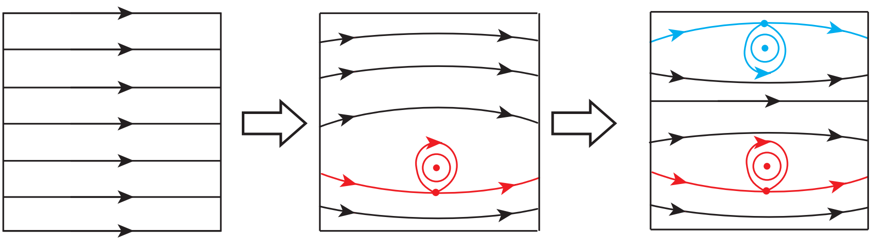
4. Topological str and Data str of surface flows

Uniform flows with vortices $\Longrightarrow$ Reduced tree structures

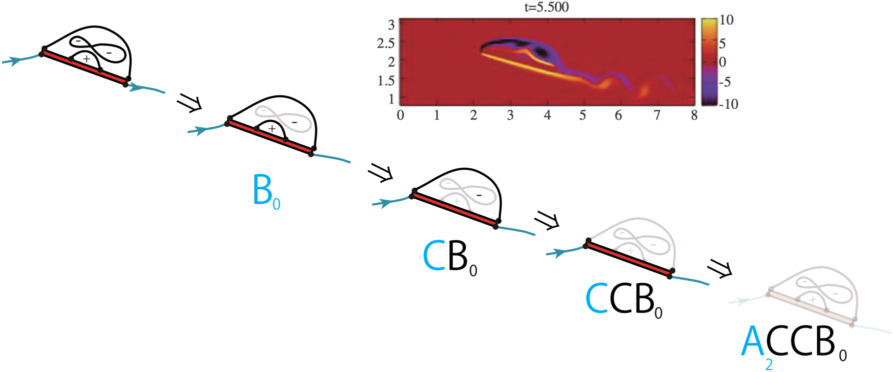
$$W(z) = Uz \implies \varepsilon$$

$$W(z) = Uz + i\kappa \log z \implies A_0$$

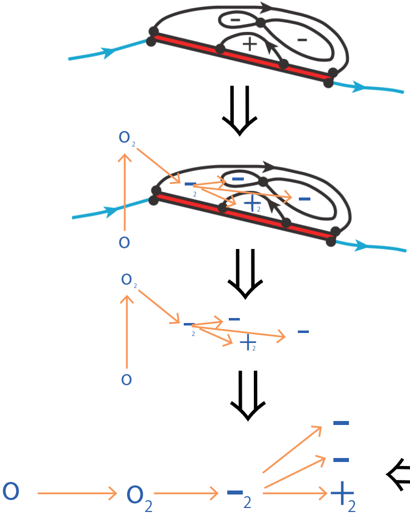
$$W(z) = Uz + i\kappa \log z - i\kappa \log(z-a) \implies A_0 A_0$$

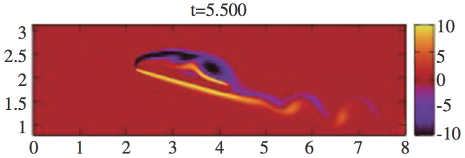


Uniform flows with vortices $\Longrightarrow$ Reduced tree structures



Uniform flows with vortices $\Longrightarrow$ Tree structures





Word representation
(Reduced tree representation)

$$AC_2CB_0$$

$$\Uparrow$$

$$\Leftrightarrow \quad o(o_2, -_2(-, -, +_2))$$

Tree representation

Tree representation (Joint work with T. Sakajo)

We construct algorithmically an injection:

$$\begin{array}{c}\exists \text{ inj} : \{\text{generic unif flow w/ vortices}\}/\sim \\ \downarrow \\ \{\text{tree}\} \\ \wr\| \\ \{\text{regular expression}\}\end{array}$$

The injection is called a tree representation of uniform flows with vortices

Uniform flows $\Longrightarrow$ Tree structures

Local stream topological structure $\Longrightarrow$ Letter

Global stream topological structure $\Longrightarrow$ Label + Edge

Topology of a flow $\Longrightarrow$ Labelled graph

Topology of a time-dep flow $\Longrightarrow$ Seq of labelled graphs

1. Applications and Background
2. Rep of uniform flows with pt vortices
3. Rep of surface flows
4. Topological str and Data str of surface flows

Surface flows $\Longrightarrow$ Tree structures

A flow is of finite type $\overset{\text{def}}{\Longleftrightarrow}$ 1. Each stagnation point is non-degenerate

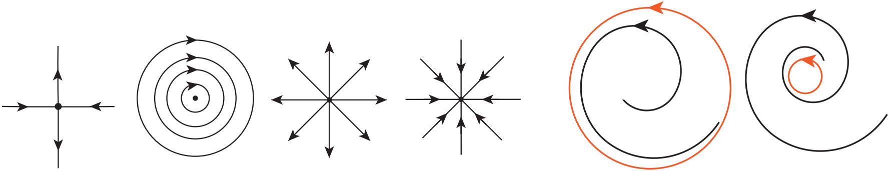
2. There are at most finitely many limit cycles

and 3. There are no quasi-minimal sets

Theorem 1

A flow of finite type on a compact surface can be reconstructed by finite data.

quasi-minimal set = orbit closure of a non-closed recurrent point

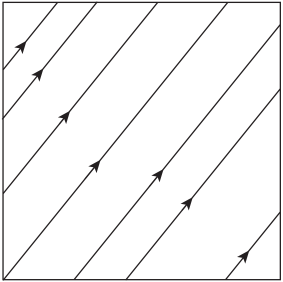


saddle center source sink Limit cycles

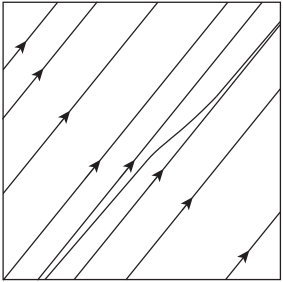
Necessity of the condition 3 (Non-existence of quasi-minimal sets)

Remark 1

The set of topological equivalent classes of minimal flows (resp. Denjoy flows) on a torus is uncountable.



Minimal flow



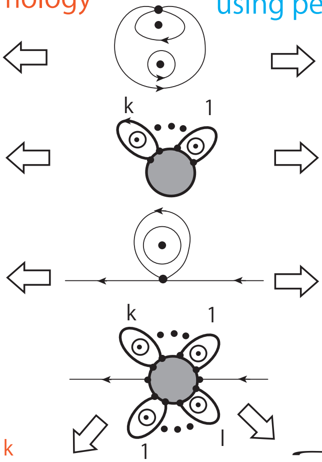
Denjoy flow

1. Applications and Background
2. Rep of uniform flows with pt vortices
3. Rep of surface flows
4. Topological str and Data str of surface flows

Good data str and Bad data str (Joint work with T.Yokoyama)

Hard to compute by computers using persistent homology

Easy to compute by computers using persistent homology



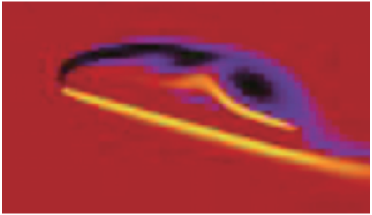
$+(+,-)$ ⇐ ⇒ β_{i+}

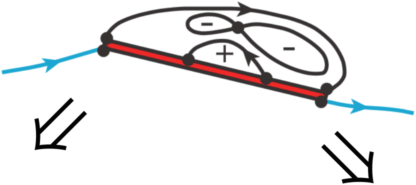
$(\overbrace{+_2, \dots, +_2}^{k})$ ⇐ ⇒ $\beta_{2+}(\overbrace{c_+, \dots, c_+}^{k})$

$(o, +)$ ⇐ ⇒ a_+

$(o_2(\overbrace{-_2, \dots, -_2}^{l}), \overbrace{+_2, \dots, +_2}^{k})$ ⇙ ⇘ $a_2(\overbrace{c_+, \dots, c_+}^{k}, \overbrace{c_-, \dots, c_-}^{l})$

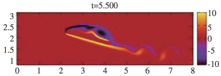
Good data structures and Bad data structures



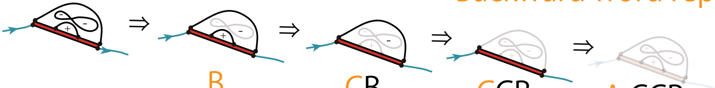


$o(o_2, -_2(-, -, +_2))$ $\alpha_2(c_-(\beta_{o-}, c_+))$

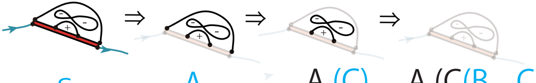
Good data structures and Bad data structures



Backward word rep



B_0 $\quad$ CB_0 $\quad$ CCB_0 $\quad$ A_2CCB_0



ε $\quad$ A_2 $\quad$ $A_2(C)$ $\quad$ $A_2(C(B_0, C))$

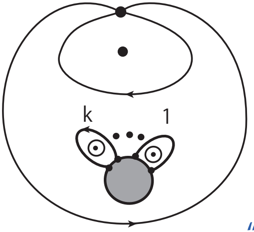
Forward word rep

ε $\quad$ A_2 $\quad$ A_2C $\quad$ A_2CB_0C

Reduced forward word rep

Good data structures and Bad data structures

Data str of topology of a Hamiltonsurface flow
= Regular tree grammar + Cyclic order



cyclic order

$$\beta_{i+}(\beta_{2+}(\overbrace{c_{+}, \dots, c_{+}}^{k}))$$

“Regular tree grammar”

Summary

We construct the following injective representation concretely

$$\mathrm{inj} : \{\text{ Uniform flow with vortices }\} \to \{\text{ finite labelled graphs }\}$$

We show the existence of following injective representation

$$\mathrm{inj} : \{\text{ surface flow of finite type}\} \to \{\text{ finite data }\}$$

Surface flows and Data structures

Topology of a flow $\Longrightarrow$ Regular tree grammar $+$ Cyclic order
Resolution $\Longrightarrow$ Depth of nodes
Good data structure $\Longrightarrow$ Persistent $\Longrightarrow$ Easy to implement
Bad data structure $\Longrightarrow$ Sensitive to error $\Longrightarrow$ Hard to implement

Appendix

Idea

Spatial setting

Rough representation : Space $\mapsto$ Homotopy group

Finer representation : Space $\mapsto$ Cell decomposition

We construct analogical tools as follows:

Flow setting

Rough representation : Hamiltonian flow $\mapsto$ Word

Finer representation : Flow $\mapsto$ Graph

Uniform flows $\implies$ Tree structures

vortex + orientation + (un)bdd-ness $\implies$ Letter

(ss-)saddle connection diagram $\implies$ Label + Edge

Snapshot of a unif flow w/ vortices $\implies$ Labelled graph

Uniform flow w/ vortices $\implies$ Seq of labelled graphs

Surface flows and Data structures

Topology of a flow	$\Longrightarrow$	Regular tree grammar + Cyclic order		
Resolution	$\Longrightarrow$	Depth of nodes		
Good data structure	$\Longrightarrow$	Persistent	$\Longrightarrow$	Easy to implement
Bad data structure	$\Longrightarrow$	Sensitive to error	$\Longrightarrow$	Hard to implement