

CoolingGen

Eine Software zur Erstellung von Kühlungsgeometrien

Julian Lüken
30. September 2022



Wissen für Morgen



Inhalt

- ▶ Einleitung
- ▶ Motivation
- ▶ Ergebnisse
- ▶ Methoden
- ▶ Fragen



Einleitung/Motivation

Was ist CoolingGen?

- ▶ Programm, welches mithilfe von CAD eine Schaufel aus BladeGen mit Kühlungsgeometrien ausstattet
- ▶ Basiert auf BasicTools (Bibliothek vom DLR für B-Spline Kurven/Flächen)
- ▶ Entwicklung startete 2013 (Autoren: C. Voß, T. Schu...)
- ▶ Meine Arbeit daran startete im Juli 2021

Warum CoolingGen?

- ▶ Erzeugung von Kühlungsgeometrien innerhalb einer Schaufel mit herkömmlichen CAD-Tools ist mühsam und dauert lange
- ▶ Laufzeit von CoolingGen: ca. 20 Sekunden auf 8 · 3GHz
- ▶ Ermöglicht Optimierung von Kühlungsgeometrien durch Phasenraumsuche gekoppelt mit CFD-Simulationen



Einleitung/Motivation

Input:

- ▶ Schaufelgeometrie aus BladeGen
- ▶ Parameter für die Kühlungsgeometrien (als XML)

Output:

- ▶ Kühlungsgeometrien (als STEP, für CENTAUR und für Tecplot)

Welche Geometrien kann CoolingGen erzeugen? Derzeit unterstützt (und hier vorgestellt):

- ▶ Kühlkanäle (mit Umkehrungen)
- ▶ Prallbleche (mit Bohrungen)
- ▶ Filmkühlung

To-do:

- ▶ Ausblasungsschlitze gut machen
- ▶ Pin-fins (Kühlrippen)



Ergebnisse

<Kanal mit Umlenkungen und FK Bohrungen>

<Kanal mit Impingement
Inserts>



Methoden

Größte Herausforderung? Robuste und performante Verschnittalgorithmen!

The single greatest cause of poor reliability of CAD systems is lack of topologically consistent surface intersection algorithms.

– Closing the Gap Between CAD Model and Downstream Applications by R. Farouki



Methoden / Kanäle und Prallbleche

Um die Kanäle zu erzeugen, werden m Kanalwände (in Relation zur Skelettlinie) als Input spezifiziert. Es wird folgende Strategie verwendet:

1. Schaufeloberfläche in radialer Höhe n mal sampeln $\rightarrow n$ Profilkurven
2. Koordinatentransformation $(x, y, z) \rightarrow (x, r) \rightarrow (m', \theta)$
3. *Schrumpfen* der Profilkurven
4. *Unterteilung* der Profilkurven an den Kanalwänden $m \rightarrow n(m + 1)$ Kammerschnitte
5. *Schrumpfen* der Kammerschnitte
6. Einpassung von *Fillets* an *Knicken/Ecken*
7. Rücktransformation nach (x, y, z) , und *Lifting*, um $m + 1$ 3D Oberflächen zu erhalten



Methoden / Kanäle und Prallbleche / Schrumpfen

Eine Offset-Kurve von γ mit Abstand d ist gegeben durch

$$O_d^\gamma(t) := \gamma(t) + dN^\gamma(t)$$

wobei $N^\gamma(t)$ der Normalenvektor von $\gamma(t)$ ist.



Wir hätten die Kurve O_d^γ gerne injektiv. Das gilt jedoch nur stückweise \rightarrow Trimming an Selbstschnittpunkten. Das Trimming hinterlässt nicht diff'bare Stellen (Knicke) \rightarrow Fillets!

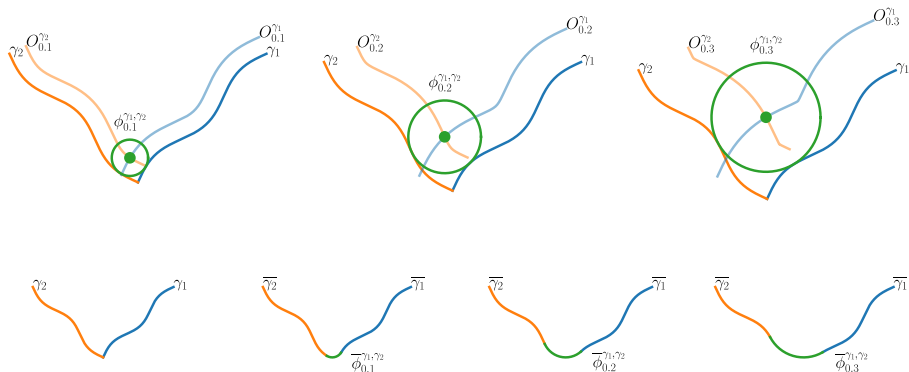


Methoden / Kanäle und Prallbleche / Fillets

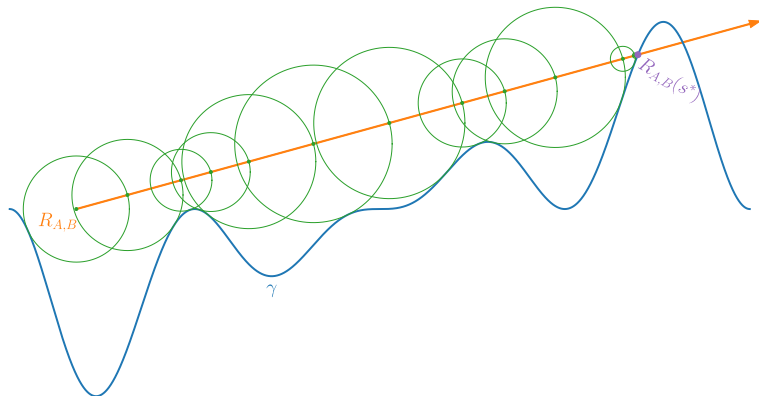
Gegeben: Radius r , Kurven γ_1, γ_2 . Man kann zeigen: Falls ein Filletkreis existiert, dann beschreibt

$$O_r^{\gamma_1}(t) = O_r^{\gamma_2}(s)$$

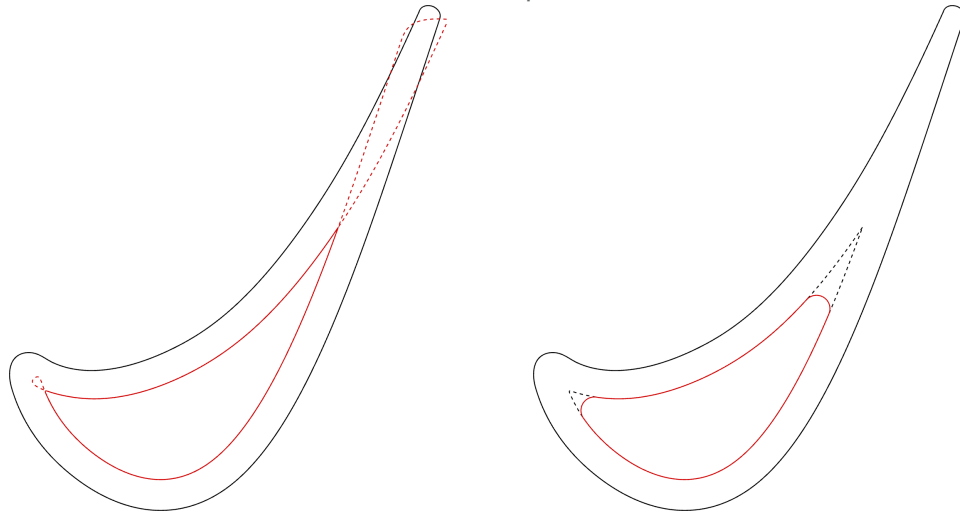
den Mittelpunkt (bzw. die Mittelpunkte). Beweis gibt es in meiner Masterarbeit.



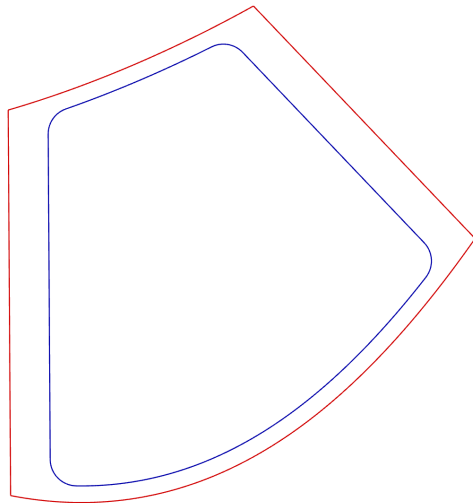
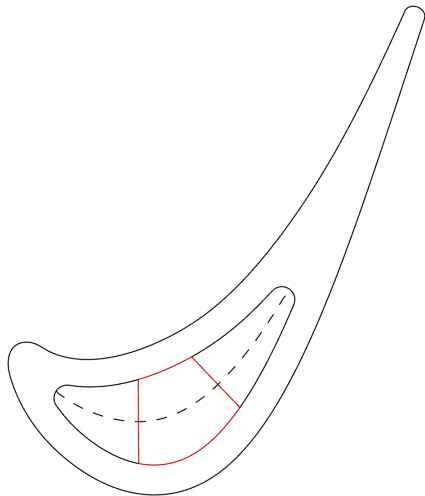
Methoden / Kanäle und Prallbleche / Ray-Marching



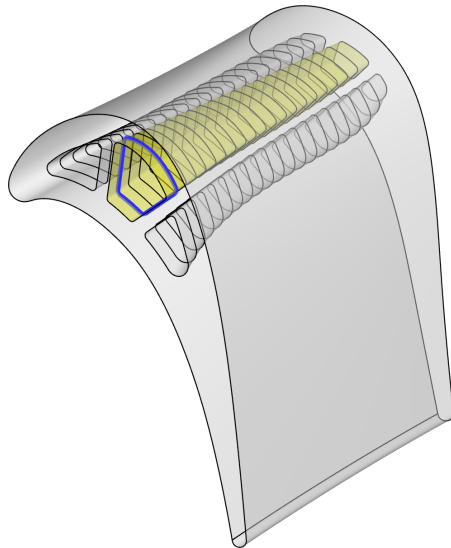
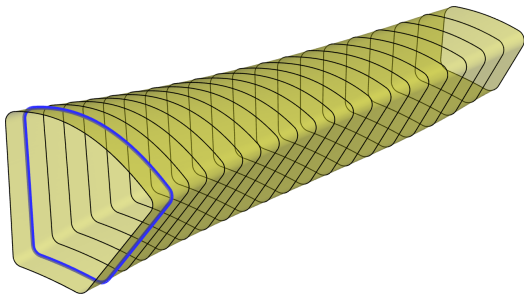
Methoden / Kanäle und Prallbleche / Schrumpfen der Profilkurven



Methoden / Kanäle und Prallbleche / Unterteilung



Methoden / Kanäle und Prallbleche / Lifting



Methoden / Umkehrungen

Strategie:

1. Berechnung einer Kombination aus den zwei Kanälen, die sich zur Umkehrung verbinden
2. Schneiden der kombinierten Kanäle mit n Ebenen $\rightarrow n$ planare Profilkurven
3. Verformen der n Kurven
4. Lifting



Methoden / Umkehrungen

<Bild aus Gitlab>

<Bild einer Turnfläche>



Methoden / Bohrungen

<Schema der Kurven einer FK Bohrung>

<Bild von Robin von Bohrung mit Params>



Methoden / Fazit

Man braucht robuste, schnelle
Verschneidungsalgorithmen, und einer reicht nicht:

1. Punkt/Kurve (um den passenden Parameter auf der Kurve zu finden)
2. Ray/Kurve
3. Kurve/Kurve und Kurve mit sich selbst
4. Fläche/Ebene
5. Fläche/Kurve

Überdies wird eine bestimmte Gestalt dieser Schnitte angenommen, um die Algorithmen effizient zu gestalten. Beispielsweise wird bei Kurve/Kurve die Annahme getroffen, dass sich diese beiden nur an einer endlichen Vereinigung von einelementigen Punktmengen treffen. Das ist zwar ausreichend, aber nicht der allgemeine Fall. Für den allgemeinen Fall gibt es gar keine "richtige" Lösung.



Methoden / Umkehrungen

<Bild aus Gitlab>

<Bild einer Turnfläche>



Recap

Kopie der Folie mit den Ergebnissen



Fragen/Anmerkungen?

