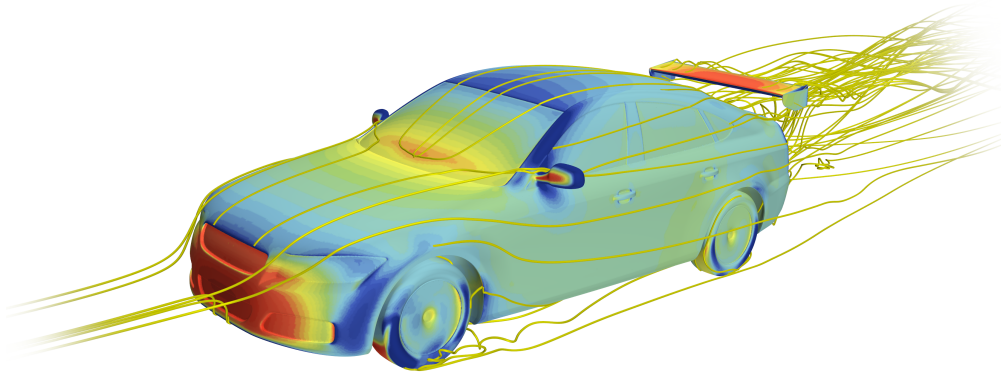




CHALMERS
UNIVERSITY OF TECHNOLOGY



Strategier för aerodynamisk formoptimering av bakvingen på en generisk tävlingsbil

Kandidatarbete inom Mekanik och maritima vetenskaper

Sebastian Eliasson Nilsson
Henrik Helmfrid
Daniel Hård
Johan Högman
Albin Knutsson
Max Rexmo

KANDIDATARBETE INOM MEKANIK OCH MARITIMA VETENSKAPER

Strategier för aerodynamisk formoptimering av bakvingen på en generisk tävlingsbil

Sebastian Eliasson Nilsson
Henrik Helmfrid
Daniel Hård
Johan Högman
Albin Knutsson
Max Rexmo



CHALMERS
UNIVERSITY OF TECHNOLOGY

Institutionen för Mekanik och maritima vetenskaper
Avdelningen för fordonsteknik och autonoma system
CHALMERS TEKNISKA HÖGSKOLA
Göteborg, Sverige 2019

Strategier för aerodynamisk formoptimering av bakvingen på en generisk tävlingsbil

- © Sebastian Eliasson Nilsson, 2019
- © Henrik Helmfrid, 2019
- © Daniel Hård, 2019
- © Johan Högman, 2019
- © Albin Knutsson, 2019
- © Max Rexmo, 2019

Handledare: Adam Brandt, Institutionen för mekanik

Handledare: Emil Ljungskog, Institutionen för mekanik

Examinator: Prof. Simone Sebben, Institutionen för mekanik

Kandidatarbete 2019:19

Institutionen för Mekanik och maritima vetenskaper

Avdelningen för fordonsteknik och autonoma system

Chalmers tekniska högskola

SE-412 96 Göteborg

Telefon 031 772 1000

Försättsblad: Strömlinjer med visualiserat tryck över bilens yta för en generisk GT3-bil med vinge.

Typsättning i L^AT_EX
Göteborg, Sverige 2019

Sammandrag

Formoptimering inom fordonsaerodynamik har traditionellt sett skett genom *trial and error*, eller med hjälp av tester i vindtunnlar. Inom GT3-racing är det viktigt att få så mycket negativ lyftkraft som möjligt utan att öka luftmotståndet för mycket. Med datorers ökade beräkningskapacitet har numeriska simuleringar blivit allt mer vanligt och därför finns det intresse att undersöka hur olika optimeringsmetoder lämpar sig för fordonsaerodynamik. Denna rapport syftar till att utvärdera tre optimeringsmetoder, på den inom forskning väl använda DrivAer-modellen, för att optimera position och attackvinkel på en bakvinge. De optimeringsmetoder som undersökts i projektet är surrogatmodellering, genetiska algoritmer och gradientbaserad formoptimering av adjointa fält.

Surrogatmodellen genomfördes med två optimeringsvariabler, höjd och attackvinkel, för att senare i projektet kompletteras med en tredje variabel, vingens position i longitudinellt led. Med ett få antal simuleringar konvergerade höjden och attackvinkel mot optimala värden för att minska luftmotståndskoefficienten, C_D , med en bibehållen lyftkoefficient, C_L . Resultaten visar att modellen optimerar två variabler snabbt trots en stor lösningsrymd.

Den genetiska algoritmen har visat sig vara mycket beräkningstung, då det krävs många generationer för att med säkerhet uppnå ett optimalt resultat. Metoden verkade konvergera mot ett visst värde. Studien är dock för kort för att veta om det var ett globalt minimum. Den genetiska algoritmen faller kort jämfört med surrogatmodellen då utvärderingen av varje individ är för beräkningstung för att göra den genetiska algoritmen effektiv.

Den gradientbaserade formoptimeringen med adjointa fält skiljer sig i sitt utförande jämfört med surrogatmodellen och den genetiska algoritmen. Dessa metoder optimerar utifrån variabler medan adjointmetoden istället optimerar genom att förändra geometrin. Adjointmetoden visade sig ha svårt att konvergera för detta fall. Optimering med en stegfunktion fungerade inte, men användandet av en sigmoid funktion var mer lovande. För att utnyttja adjointmetoden effektivt ansågs det vara bra att använda surrogatmodellering eller genetiska algoritmer tillsammans med adjointmetoden.

Nyckelord: adjointmetoden, genetisk algoritm, surrogatmodell, DrivAer, bakvinge, luftmotståndskoefficienten (C_D), lyftkoefficienten (C_L)

Abstract

Traditionally, shape optimization in vehicle aerodynamics have been done through trial and error, or with facilitation of wind tunnels. In GT3 racing it is important to generate as much downforce (negative lift force) as possible without any significant increase of drag. With increasing computational power, numerical simulations have become more common. Thus it is of interest to evaluate how applicable different optimization methods are in vehicle dynamics. This report aims to evaluate three optimization methods on the scientifically commonly used DrivAer model to optimize position and angle of attack of a rear wing. The optimization methods evaluated are surrogate modelling, genetic algorithms and gradient-based shape optimization using adjoint fields.

The surrogate model was implemented with two optimization variables, height and angle of attack, with a later complement of a third variable, longitudinal placement of the wing. With just a few simulations the height and angle of attack converged toward optimal values with the purpose of minimizing the drag coefficient, C_D , whilst maintaining the lift coefficient, C_L . The results shows that this optimization method optimizes two variables in a short period of time, despite a large sampling space.

The genetic algorithm has been found to be highly demanding in computational power, since it is necessary to evaluate many generations to be sure of an optimal result. It seemed to converge toward a certain value, but this study is too short to know if it is a global minimum. The genetic algorithm falls short compared to the surrogate model since the evaluation of each individual is too demanding in computational power to make it efficient enough.

The gradient based optimization method using adjoint fields differs from the surrogate model and the genetic algorithm in its execution. These methods optimizes input variables whilst the adjoint method optimizes by changing the geometry. The adjoint method had a problem with convergence in this application. Optimization with a step function did not work, but instead using a sigmoid function seemed more promising. To use the adjoint method effectively it should be a complement to either the surrogate model or the genetic algorithm.

Keywords: adjoint method, genetic algorithm, surrogate model, DrivAer, rear wing, drag coefficient (C_D), lift coefficient (C_L)

Förord

Kandidatarbetet "Strategier för aerodynamisk formoptimering av bakvingen på en generisk tävlingsbil" löper under vårterminen år 2019 på Chalmers tekniska högskola, Göteborg. Projektet utförs av sex studenter varav alla läser sitt tredje år på olika civilingenjörsprogram på Chalmers. Utbildningsbakgrunden varierar mellan maskinteknik och teknisk fysik. Valet att utföra detta arbete grundar sig i ett gemensamt intresse för mekanik, strömningsmekanik och tillämpningar av dessa ämnen.

Arbetet inleds med att tacka våra handledare Adam Brandt och Emil Ljungskog för en mycket bra och välskött handledning av detta arbete. Vidare vill vi tacka vår examinator Prof. Simone Sebben för att hon givit oss möjligheten att utföra detta arbete. Vi vill även tacka doktorand Magnus Urquhard och doktorand Sina Torabi för hjälp med diverse programmeringsskript tillämpade i detta kandidatarbete. Slutligen tackar vi Institutionen för Mekanik och maritima vetenskaper för tillhandahållning av diverse nödvändiga resurser för utförande av detta kandidatarbete.

Författarna, Göteborg, maj 2019

Innehåll

Nomenklatur	xii
Figurer	xiv
Tabeller	xv
1 Inledning	1
1.1 Bakgrund	1
1.2 Syfte och mål	2
1.3 Avgränsningar	2
2 Teori	3
2.1 CFD - numeriska fluidberäkningar	3
2.1.1 Finita volymmetoden	3
2.1.2 $k - \omega$ SST modellen	4
2.2 Strömningsteori	4
2.3 Fordonsaerodynamik	5
2.3.1 Negativ lyftkraft och strömning kring vinge	6
2.3.2 Luftmotstånd	8
2.4 Formoptimeringsmetodik	8
2.4.1 Surrogatmodellering	8
2.4.1.1 Latin hypercube sampling	9
2.4.1.2 Radial basis function	9
2.4.1.3 Bedömning av resultat	10
2.4.2 Genetiska algoritmer	10
2.4.3 Adjointmetoden	12
2.4.3.1 Adjointekvationen	13
3 Metod	15
3.1 Förarbete	15
3.1.1 DrivAer-modellen utan bakvinge	15
3.1.2 Konstruktion av bakvinge	16
3.1.3 Beräkningsdomän	17
3.1.4 Fysikalisk modell	18
3.1.5 Solver-inställningar	18
3.1.6 Mesh	19
3.1.6.1 Meshgenerering	19

3.2	Optimering	21
3.2.1	Latin hypercube sampling	22
3.2.2	Konvergens	24
3.2.3	Surrogatmodellering	25
3.2.4	Genetiska algoritmer	26
3.2.5	Adjointmetoden	27
3.2.5.1	Kostnadsfunktion	28
3.2.5.2	Konvergens av adjointlösaren	29
3.2.5.3	Kontrollpunkter och deformation av geometri	30
4	Resultat	31
4.1	Latin hypercube sampling	31
4.2	Surrogatmodellering	32
4.2.1	Strömlinjer och vak	36
4.3	Genetiska algoritmer	36
4.3.1	Strömlinjer	39
4.4	Adjointmetoden	40
4.4.1	Kostnadsfunktion	40
4.4.2	Konvergens	40
4.4.3	Ytkänslighetsdata	41
5	Diskussion	43
5.1	Surrogatmodellen	43
5.1.1	Utvärdering och rekommendationer	44
5.2	Genetiska algoritmer	44
5.2.1	Mutering	45
5.2.2	Utvärdering och rekommendationer	45
5.3	Jämförelse mellan surrogatmodellering och genetiska algoritmer	46
5.3.1	Strömlinjer och vak	46
5.3.2	Luftmotståndskoefficienten, C_D	46
5.3.3	Summering	47
5.4	Adjointmetoden	47
5.4.1	Utvärdering och rekommendationer	49
6	Slutsatser	51
	Litteraturförteckning	55
A	Förarbete	I
B	Surrogatmodellen	III
C	Genetiska algoritmen	VII
D	Adjointmetoden	IX

Nomenklatur

Programvara

ANSA	Programvara för förbearbetning av geometrier
Julia	Programmeringsspråk
MatLab	Programmeringsspråk
STAR-CCM+	Programvara för CFD

Förkortningar

CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
Curve-fit	Konstruktion av matematisk funktion som passar bäst i en serie av punkter
DOE	Design of experiments
FVM	Finita Volymmetoden
GMRES	Generalized Minimal Residual Method
LHS	Latin hypercube sampling
MEDIAN	Bästa värdet för optimeringsvariablerna, givet av surrogatmodellen
RAND	Randomiserade värden på optimeringsvariablerna, givet av surrogatmodellen
RANS	Reynolds-averaged Navier-Stokes
RBF	Radial basis function
SST	Shear Stress Transport
STD	Standardavvikelse i surrogatmodellen
Steady-state	Stabilt tillstånd - Tidsberoende variabler

Symboler

ρ	Densitet [$\frac{\text{kg}}{\text{m}^3}$]
u	Hastighet [$\frac{\text{m}}{\text{s}}$]
F_L	Lyftkraft [N]
F_D	Dragkraft [N]
p	Tryck [$\frac{\text{N}}{\text{m}^2}$]
t	Tid [s]
w	Viktfunktion beroende av den radiella basfunktionen
Φ	Radiell basfunktion
Υ	Matris innehållande basfunktioner
Λ	Adjointa fältet
μ	Viskositet [$\frac{\text{kg}}{\text{m s}}$]
ν	Kinematisk viskositet [$\frac{\text{m}^2}{\text{s}}$]
u^*	Dimensionslös hastighet relativt referenshastighet [-]
y	Avstånd från vägg [m]

Figurer

2.1	Dimensionslös hastighetsprofil för ett gränsskikt vid turbulent flöde [7].	5
2.2	Trycket är högre på ovansidan av vingen än på undersidan, vilket resulterar i att vingen dras nedåt.	6
2.3	En vinge med en liten attackvinkel som har liten separation. Flödeslinjerna återförenas efter att ha passerat vingen. Endast lite turbulens bildas efter vingen, i bilden representerat av de två skarpt böjda kurvorna efter vingen.	7
2.4	Separation sker på en vinge vid höga attackvinklar. Det turbulenta gränsskiktet är markerat direkt bakom vingen. I denna region kan återcirkulation ske.	7
2.5	Flödesschema för surrogatmodellering.	9
2.6	Uppbyggnad av en individs genom.	11
2.7	Tillvägagång för att skapa ett barn från två individer.	11
2.8	En individ före och efter mutation.	12
3.1	Strömlinjer för DrivAer-modellen utan vinge.	16
3.2	NACA-profil, S1223 - Selig S1223 high lift low Reynolds number airfoil [20].	16
3.3	Den konstruerade vingen för användning till optimeringsmetoderna.	17
3.4	Dimensioner för beräkningsdomänen. Notera att domänen endast inkluderade halva bilen, då den är delad längs bilens symmetriplan.	17
3.5	Förfining av volymmesh genom stegvis minskande <i>Density Boxes</i> runt bilen, med en minskad cellstorlek i varje. Inställningar för dessa presenteras i tabell 3.6.	19
3.6	Attackvinkel noll grader för vingen sker när vingen och bakrutan har samma vinkel.	21
3.7	Vinge relativt lokala koordinatsystemet på bakluckan.	21
3.8	Intervall för höjd och vinkel. Denna figur visar också tydligt att vingens sidopaneler följer med vid vinkeländring av vingprofilen.	22
3.9	Latin hypercube sampling. Notera att enheten för höjd är i centimeter.	23
3.10	Residualer för en simulering ligger mellan $5 \cdot 10^{-3}$ och $5 \cdot 10^{-5}$	24
3.11	Visualisering av y^+ kring vingen.	24
3.12	Absoluta värdet samt medelvärdet för C_D och C_L . C_D värdet har konvergerat bra, medan C_L fluktuerar. Dock är fluktuationen tillräckligt jämn och därför anses medelvärdet ha konvergerat.	25
3.13	Uppbyggnad av verkligt genom för en individ.	26

3.14	Arbetsprocessen vid användning av adjointmetoden.	27
3.15	Funktionen f i (3.1) och en stegfunktion mellan C_D och $2 \cdot C_D$, med diskontinuiteten placerad vid $C_L = -0,287$	29
4.1	Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L från LHS, för både surrogatmodellen och genetiska algoritmen, med $C_D = 0,347$ och $C_L = -0,424$	31
4.2	Konturplottar för optimeringsvariabler i korrelation med C_D och C_L	33
4.3	Konturplottar för optimeringsvariabler i korrelation med C_D och C_L	34
4.4	Strömlinjer för simulering 37 och 39.	36
4.5	Hastighetsprofil för simulering 37 och 39.	36
4.6	C_D och C_L för den bästa individen varje generation.	37
4.7	Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation 2.	37
4.8	Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation 4.	38
4.9	Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation åtta.	38
4.10	Konturplottar för optimeringsvariabler i korrelation med C_D och C_L	39
4.11	Visualisering av strömningslinjer kring vingen med lägst C_D som klarar kravet på C_L	39
4.12	Residualer för 100-10 och <i>1st-order</i> . Notera att vid iteration 2011 ökar residualerna snabbt för att sedan minska igen vilket skedde när adjointlösaren fortsatte.	41
4.13	Ytkänsligheten för vingen med avseende på ekvation (3.1). Notera att bara en liten del av vingen ska förändras enligt beräkningarna.	42
5.1	Jämförelse för konvergens av C_D , givet kravet på C_L , för respektive metods dittills bästa simulering.	47
A.1	Ritning för vingen.	I
B.1	Residualer för simulering 39 med surrogatmodellen.	IV
B.2	y^+ på bilen för simulering 39 med surrogatmodellen.	V
B.3	Hastighetsprofil för simulering nummer 34 med annat krav på C_L	VI
D.1	Residualer för <i>Restarted GMRES</i> 50-3.	IX
D.2	Residualer för <i>Flexible GMRES</i> 50-3.	X
D.3	Residualer för <i>Flexible GMRES</i> 100-10 och diskretisering på <i>2nd-order</i>	X

Tabeller

2.1	Typ av RBF.	10
3.1	Resultat för DrivAer-modellen utan bakvinge.	15
3.2	Randvillkor för beräkningsdomän med DrivAer-bil.	18
3.3	Fysikaliska modellen för CFD.	18
3.4	Solverinställningar i STAR-CCM+.	19
3.5	Inställningar för den genererade meshen.	20
3.6	Inställningar för <i>Density Boxes</i> som är illustrerade i figur 3.5.	20
3.7	Inställningar för <i>Surface Control</i> på vingens yta.	20
3.8	Värden från LHS.	23
4.1	Konvergensintervall för surrogatmodellering.	32
4.2	Gränser för intervallet.	32
4.3	Resultat från surrogatmodellens tio sista simuleringar. De bästa simuleringarna är markerade med ljusgrön färg.	35
5.1	Jämförelse av luftmotståndskoefficienten hos surrogatmodellen och den genetiska algoritmen.	46
B.1	Alla simuleringar för surrogatmodellen.	III

1

Inledning

1.1 Bakgrund

Inom motorsport är målet att ta sig fram så fort som möjligt. Därför eftersträvas att alltid hålla så höga hastigheter som möjligt. Större negativ lyftkraft ger större krafter mellan däck och väglag vilket leder till att bilen klarar högre lateral acceleration genom en kurva utan att däcken tappar fästet. Kurvtagning kan således göras i högre hastighet med ökat negativt lyft. Anledningen till att aerodynamik inom motorsport är ett så komplext ämne är att det uppstår en avvägning mellan maximal negativ lyftkraft och minimalt luftmotstånd. Högre negativ lyftkraft bidrar till bättre kurvtagning medan mindre luftmotstånd bidrar till högre maxhastighet. En större negativ lyftkraft sker oftast till priset av ett högre luftmotstånd, vilket sänker maxhastigheten [1].

Både luftmotståndet och den negativa lyftkraften beror på många olika variabler. Till exempel kan en ökning av vinkeln på vingen gentemot luftströmmen skapa högre negativ lyftkraft. Även variabler som höjd gentemot bakpartiet på bilen, bredd på vingen och olika typer av skalningar av vingen har en påverkan på hur hög negativ lyftkraft som bildas.

För att aerodynamiskt utveckla en bil används både fysiska experiment samt numeriska beräkningar [2]. De fysiska experimenten består främst av tester i vindtunnel. En begränsande faktor är att det krävs en fysisk modell, vilken kan vara tidskrävande att ta fram. Ett alternativ är då att utnyttja numeriska beräkningar i form av CFD-simuleringar, vilka endast kräver en CAD-modell för att kunna testas. Numeriska beräkningar lämpar sig även bra för det optimeringsproblem som uppstår mellan negativ lyftkraft och luftmotstånd eftersom man relativt enkelt kan studera och ändra relevanta geometriska parametrar. På så vis kan mycket av optimeringen bli automatiserad, snarare än att utnyttja en process av *trial and error*.

1.2 Syfte och mål

Detta kandidatarbete syftar till att undersöka för- och nackdelar med tre olika optimeringsstrategier när de används för att aerodynamiskt formoptimera en bakvinge på en generisk GT3-bil. De olika optimeringsstrategierna är surrogatmodellering, genetiska algoritmer och adjointmetoden. Målet är att bibehålla negativ lyftkraft och sänka luftmotståndet jämfört med ett utgångsfall. I slutet av arbetet är målet att kunna redovisa hur de olika optimeringsmetoderna förhåller sig till varandra avseende beräkningsmässig kostnad och resultat vid formoptimering av en bakvinge.

1.3 Avgränsningar

Ingen fysisk modell har konstruerats för validering av genomförda simuleringar i en vindtunnel. Vid simulering är hjulen rakställda. Vidare är fluiden som används luft, strömmande vid en temperatur av 20 °C. Fluiden flödar med en konstant hastighet på 100 km/h mot modellen och simulerar därmed en konstant hastighet för bilen. Fluidhastigheten ansätts till en hög hastighet för att säkerställa att C_D -värdet har ett lågt beroende av Reynoldstalet. De termodynamiska effekterna som uppstår förväntas ha en försumbar inverkan och kommer därför inte beaktas i detta arbete.

Simuleringar genomförs på givna DrivAer-modell [3]. Modellen förenklas genom att använda stängda fälgar på grund av att inverkan av flödet kring fälgarna inte antas påverka flödet kring bakvingen märkvärt. DrivAer-modellen delas längs med symmetriplanet då det antas att flödet kring bilen är symmetriskt. Detta bör innebära att beräkningstiden för simuleringarna minskas. Simuleringen sker i *steady-state* och turbulensmodellen som därmed används är av RANS-modell, specifikt SST $k - \omega$ modellen.

Utöver ovan nämnda avgränsningar finns även begränsningar. Arbetet har en begränsad beräkningskapacitet om 10000 core-timmar på Chalmersklustret Vera per månad. Gruppen har även en tidsbegränsning om 20 arbetstimmar per vecka och person att avsätta för projektet.

Simuleringar utförs på en så kallad generisk tävlingsbil, vilket är den givna DrivAer-modellen. Med generisk menas att modellen är uppbyggd med målet att efterlikna en så stor del av alla bilar som möjligt. Projektets simuleringar genomförs därmed på en generisk modell för att de slutsatser som dras ska kunna appliceras på de allra flesta tävlingsbilar i GT3-kategorin.

2

Teori

I detta avsnitt behandlas den bakomliggande teorin för arbetet. Den metod och teori som används i moderna CFD-program består till stor del av finita volymmetoden, som här beskrivs först. Därefter presenteras den teori som behövs för att beräkna och analysera ett flödesfält, vilken kommer från strömningsmekaniken. Slutligen beskrivs teorin bakom de tre optimeringsmetoder som har använts; surrogatmodellering, genetiska algoritmer och adjointmetoden.

2.1 CFD - numeriska fluidberäkningar

För att kunna analysera den aerodynamiska formoptimeringen av bakvingen på den generiska DrivAer-modellen [3] används dedikerade CFD-program. CFD-simuleringar utförs med hjälp av Navier-Stokes ekvationer

$$\rho_0 \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \rho_0 \mathbf{g} \quad (2.1)$$

och dessa problem beräknas sedan numeriskt [2]. I följande kapitel beskrivs metodiken och de matematiska modeller som används.

2.1.1 Finita volymmetoden

FVM är en av de vanligaste diskretiseringsmetoderna för att lösa partiella differentialekvationer i moderna CFD-program. Metoden passar bra för CFD-simuleringar eftersom den är intuitiv att förstå och tillämpa [4].

Metoden bygger på diskretisering av differentialekvationer där flera antaganden och förenklingar gjorts. En lösningsdomän delas in i ett finit antal icke överlappande kontrollvolymmer som kallas för celler utgör det som senare kallas för volymmesh. Differentialekvationen integreras sedan över varje kontrollvolym och en lösning på den sökta storheten beräknas i centrum i varje cell. För att kunna lösa ekvationerna som beskriver flödet i hela domänen interpoleras sedan alla lösningar i respektive cells centroid [5]. Detta är en iterativ process som fortgår tills dess att konvergens uppnåtts.

2.1.2 $k - \omega$ SST modellen

$k - \omega$ SST modellen är lämplig att använda då $k - \omega$ inte kräver extra dämpningsfunktioner nära väggen, samt då variablerna ej är tidsberoende. SST-formen innebär att $k - \omega$ ändras till $k - \epsilon$ form utanför gränsskiktet vilket undviker problem som kan uppkomma med $k - \omega$ där [6]. Modellen används alltså då problemet är tidsoberoende samt att SST delen lämpar sig extra då separationsflöde kan uppkomma, vilket det ofta gör på en bil. Det är också ett stort utrymme som betraktas som friström i simuleringarna vilket $k - \omega$ SST modellen är lämpad för [6].

2.2 Strömningsteori

En viktig variabel att betrakta inom nästan alla sorters strömningsberäkningar är Reynoldstalet

$$Re = \frac{\rho VL}{\mu} = \frac{VL}{\nu} \quad (2.2)$$

vilket är ett dimensionslöst tal som beskriver fluidens viskösa beteende [2]. Olika geometrier och förhållanden ger olika betydelse på värdet av Reynoldstalet, men generellt kan det sägas att ett lågt Reynoldstal innebär laminärt flöde och ett högt värde innebär turbulent flöde. När Reynoldstalet är mycket stort påverkas fluidens beteende endast i liten grad vid förändring av Reynoldstalet [2].

Ett annat viktigt begrepp är

$$y^+ = \frac{yu^*}{\nu} \quad (2.3)$$

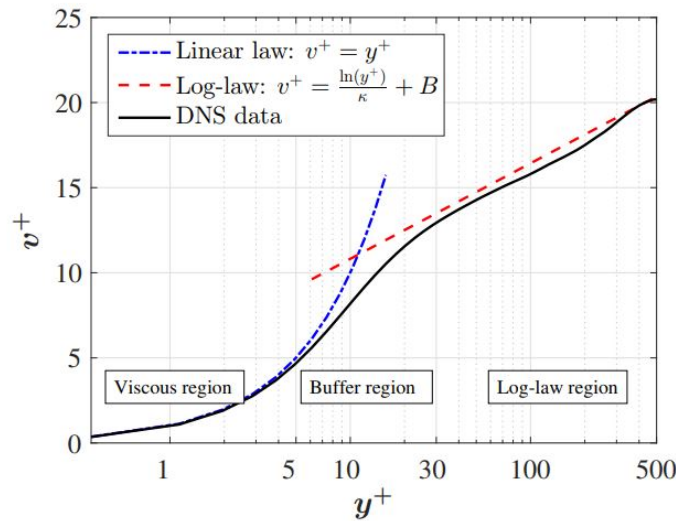
vilket är ett dimensionslöst avstånd från väggen. Eftersom tjockleken på gränsskiktet varierar i olika flödesförhållanden är y^+ lämpligt att använda inom CFD. Vid olika värden på y^+ approximeras den dimensionslösa hastigheten u^+ på olika sätt. $y^+ < 5$ kallas för det viskösa underlagret och u^+ beskrivs då av

$$u^+ = \frac{u}{u^*} = \frac{yu^*}{\nu} = y^+ \quad (2.4)$$

vilket visar att y^+ är linjärt gentemot u^+ . Vid $y^+ > 30$ är u^+ logaritmisk enligt

$$u^+ = \frac{1}{\kappa} \ln \frac{yu^*}{\nu} + B = \frac{1}{\kappa} \ln y^+ + B \quad (2.5)$$

där $\kappa \approx 0,41$ och $B \approx 5,0$. Detta kallas log-lagret [2]. Experimentell data och numeriska simuleringar, *Direct Numerical Simulation* data, DNS [7], har validerat ekvation (2.4) och (2.5) i respektive intervall. DNS datan har även använts för att beskriva hastighetsprofilen mellan dessa, kallat buffertlagret. Relationen mellan u^+ och y^+ för det viskösa underlagret, loglagret och DNS datan finns illustrerat i figur 2.1.



Figur 2.1: Dimensionslös hastighetsprofil för ett gränsskikt vid turbulent flöde [7].

2.3 Fordonsaerodynamik

Ett fordons aerodynamiska egenskaper påverkar hur stora krafter som verkar på fordonet vid färd. Den totala kraften motverkar fordonets drivkraft och kallas för fordonets luftmotstånd. Denna kraft normaliseras och ger luftmotståndskoefficienten, C_D . Med en ökande koefficient ökar energiförbrukningen.

En större användning av fossila bränslen som bensin eller diesel bidrar till stora utsläpp av koldioxid, CO_2 , i atmosfären [8]. Fordonstrafiken i Europa står för en femtedel av de totala koldioxidutsläppen inom EU och har därmed en stor påverkan [9]. Naturen kan inte ta upp de mängder av koldioxid som fordonstrafiken släpper ut. Detta leder i sin tur till att gasen lagras i atmosfären och bidrar till den globala uppvärmningen [10]. Ett optimerande av ett fordons miljöpåverkan är på så sätt något som allt fler eftersträvar.

Både i racingvärlden och i personbilsindustrin har tillverkarna som mål att optimera de aerodynamiska egenskaperna. Detta för att optimera exempelvis negativ lyftkraft, luftmotstånd eller stabilitet men även för att få en så låg bränsleförbrukning som möjligt. I racingvärlden med syftet att köra med så lite bränsle som möjligt och därmed med en lättare bil. Optimeringen utförs även för att få så stor negativ lyftkraft och lågt luftmotstånd som möjligt för att kunna uppnå högre topphastighet och högre hastigheter i kurvor. I personbilsindustrin är istället syftet att minska bränsleförbrukningen för att öka efterfrågan från kunden och att sälja en så stabil och säker bil som möjligt.

2.3.1 Negativ lyftkraft och strömning kring vinge

På ett flygplan skapas positiv lyftkraft genom användningen av vingar medan på racingbilar används samma princip för att istället generera negativ lyftkraft. Inom fordonsaerodynamiken benämns ofta negativ lyftkraft, $-F_L$, som något att eftersträva. Den negativa lyftkraften ger högre normalkraft mellan däck och väg, vilket betyder att en högre hastighet kan uppnås i kurvor. För att uppnå en negativ lyftkraft vänds vingprofilen för ett flygplan, då bildas istället ett högre tryck på ovansidan samt ett lägre tryck på undersidan vilket medför att vingen trycks nedåt istället.

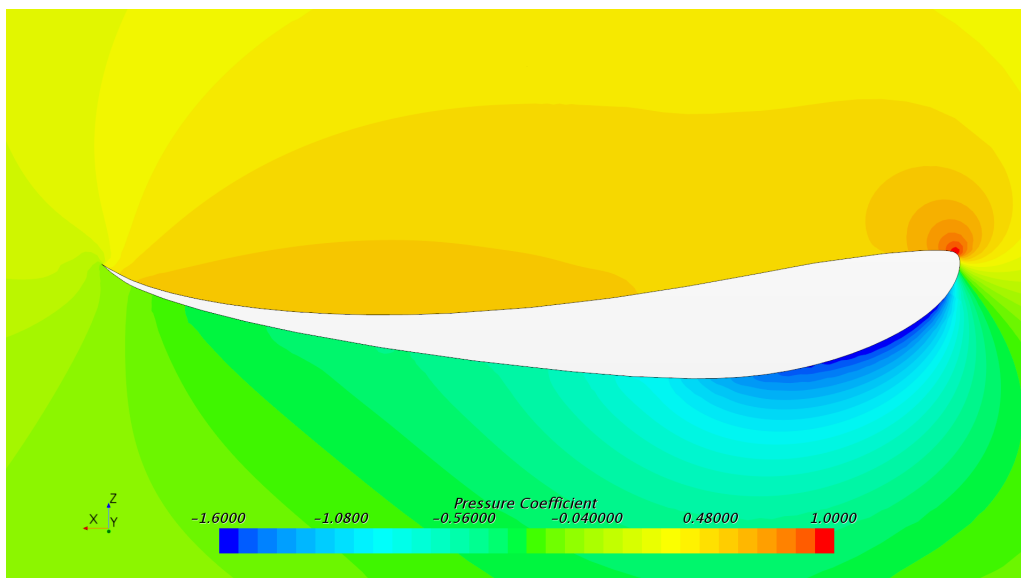
En vinge genererar negativ lyftkraft genom att påverka luftströmmen. Genom att vara mer utböjd på undersidan än översidan ökar hastigheten mer på undersidan än översidan av vingen [11]. En förenkling av Bernoullis ekvation

$$P_s + \frac{1}{2}\rho V^2 = \text{konstant} \quad (2.6)$$

visar att om hastigheten ökar så måste trycket sjunka [2]. Detta gör att ett undertryck skapas under vingen som ”drar” ner den samtidigt som ett övertryck på ovansidan av vingen trycker ner den, se figur 2.2. Ekvationen är dock (2.6) en förenkling av Navier-Stokes ekvation, se ekvation (2.1), där viskösa krafter i fluiden inte beaktas. Exempelvis vid flödesseparation gäller inte Bernoullis ekvation. Negativ lyftkraft mäts i den dimensionslösa variabeln

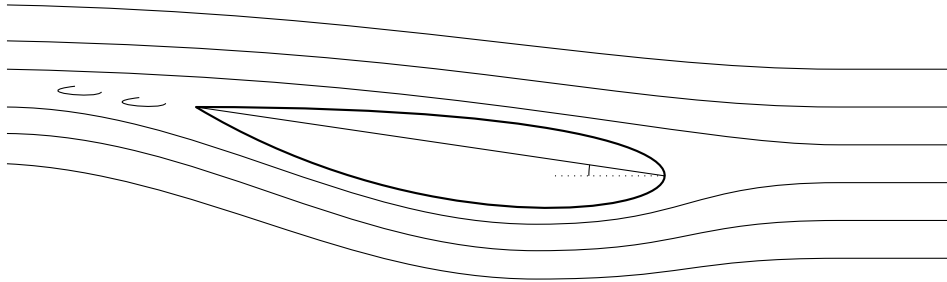
$$C_L = \frac{F_L}{\frac{1}{2}\rho V^2 A} \quad (2.7)$$

och är ett normaliserat mått på hur stora vertikala krafter som verkar på bilen [2]. Ekvationen används då den kan jämföra olika hastigheter med varandra. F_L är lyftkraften, A är den projicerade arean från ovan på vingen, ρ är fluidens densitet och V är hastigheten på friströmmen.



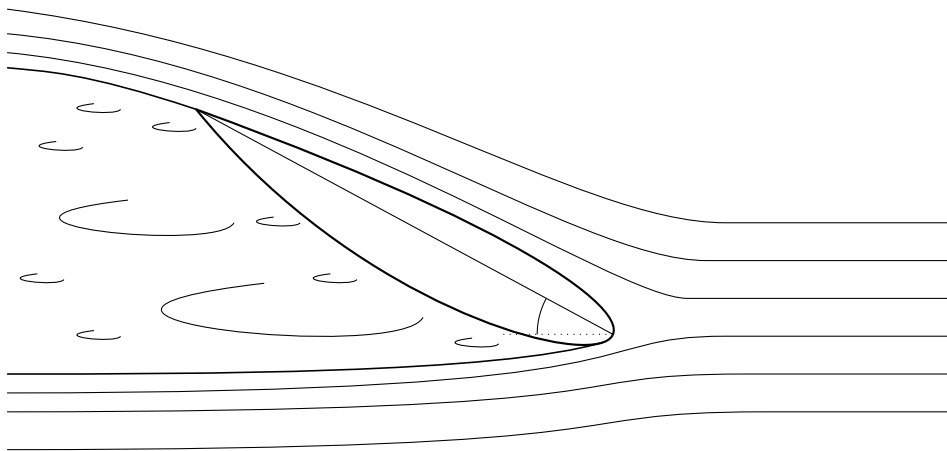
Figur 2.2: Trycket är högre på ovansidan av vingen än på undersidan, vilket resulterar i att vingen dras nedåt.

En vinge som färdas genom en fluid har en stor inverkan på flödet. Bland annat beroende på vingprofil och attackvinkel så kommer flödet att kunna följa vingens yta och skapa lyftkraft eller separera och då tappa lyftkraft. Attackvinkeln är den vinkel som vingen har mellan dess korda, linjen mellan främre och bakre spets, och den inkommande luftströmmen [2]. I figur 2.3 visas ett fall med liten attackvinkel där flödeslinjerna följer med vingprofilen och luftflödet återförenas baktill. En sådan vinge har effektiv lyftkraft gentemot luftmotståndet. Den undre ytan har en ogynnsam tryckgradient, men inte tillräckligt för att orsaka betydande separation.



Figur 2.3: En vinge med en liten attackvinkel som har liten separation. Flödeslinjerna återförenas efter att ha passerat vingen. Endast lite turbulens bildas efter vingen, i bilden representerat av de två skarpt böjda kurvorna efter vingen.

När attackvinkeln ökar kommer den ogynnsamma tryckgradienten på den undre sidan att bli starkare [2]. I figur 2.4 visas en vinge med stor attackvinkel. Luftflödet släpper helt från vingens undre yta och detta fenomen kallas för överstegring. Vid överstegring går en del av energin till turbulensbildning och därmed kan lyftkraften minska och luftmotståndet öka. Speciellt för vingen i figur 2.4 gäller att stor separation sker. Bakom vingen bildas en turbulent region där återcirkulation kan ske, vilket benämns vak.



Figur 2.4: Separation sker på en vinge vid höga attackvinklar. Det turbulenta gränsskiktet är markerat direkt bakom vingen. I denna region kan återcirkulation ske.

2.3.2 Luftmotstånd

Luftmotståndet mäts ofta i den dimensionslösa variabeln

$$C_D = \frac{F_D}{\frac{1}{2}\rho V^2 A} \quad (2.8)$$

och är ett normaliserat mått på hur mycket ett objekt motstår rörelse genom en fluid [2]. Att variabeln är dimensionslös gör att resultatet enkelt går att jämföra med andra förutsättningar, till exempel olika hastigheter eller helt andra bilar. A är den karakteristiska arean, vilket på en bil är den projicerade arean på fronten. F_D är luftmotståndet som uppstår på bilen, ρ är fluidens densitet och V är hastigheten på friströmmen relativt bilen.

Luftmotståndet uppstår huvudsakligen på grund av två fenomen, friktionsmotstånd och tryckskillnader. Dessa verkar ortogonalt mot varandra, där friktionen verkar tangentiellt med ytan och trycket verkar normalt mot ytan. För en bil uppstår ungefär 90 % av luftmotståndet på grund av tryck och 10 % på grund av friktion. Generellt gäller det att trycket ökar om fluidens hastighet är låg vid ytan, medan det minskar om hastigheten är hög, se ekvation (2.6).

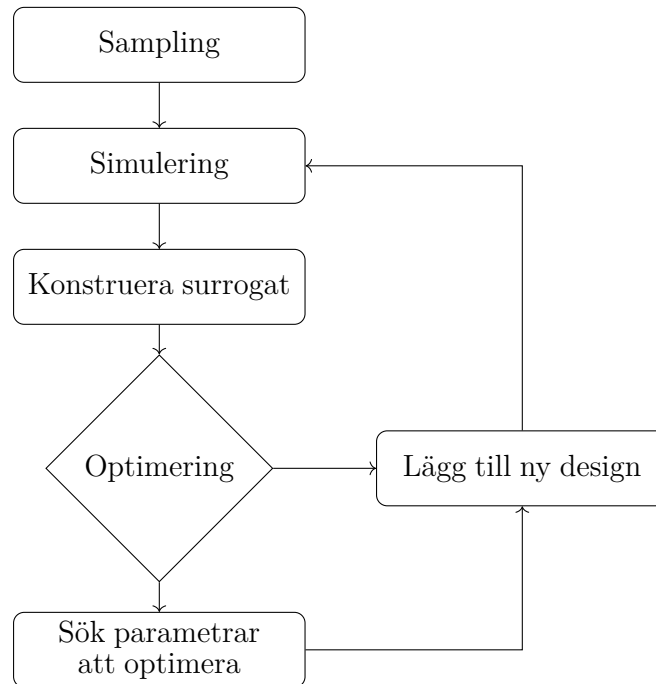
2.4 Formoptimeringsmetodik

Optimeringsmetoder går ut på att hitta den mest optimala lösningen för ett problem på ett tidseffektivt sätt. Inom fordonsaerodynamik har optimeringsförsöken länge varit *trial and error*, vilket är kostsamt och tar lång tid. Datoriserade optimeringsmetoder är därför till stor nytta inom fordonsaerodynamik. Med hjälp av optimeringsmetoderna finns det möjlighet att undersöka många variabler samtidigt och därmed ta hänsyn till hur olika variabler påverkar varandra.

2.4.1 Surrogatmodellering

Surrogatmodellering kan användas som optimeringsmetod för ett problem där det krävs många mätningar för att hitta det optimala resultatet. Istället används en approximation som ges av ett begränsat antal simuleringar för att efterlikna lösningen. Därmed reduceras ledtiden vilket gör att det blir beräkningsmässigt billigare än att simulera en fysisk modell [12]. Grundidén är att modellen tillämpar en *curve fit* för den tillgängliga data som finns. Vid en sådan tillämpning kan resultatet förutses utan att simulera hela lösningsrymden. Genom att göra ett begränsat antal simuleringar där de valda variablerna sprids ut i lösningsrymden kan en approximativ representation av hela den verkliga lösningsrymden beräknas. Under rubrik 2.4.1.1 beskrivs hur detta åstadkommes.

Den största utmaningen då en surrogatmodell konstrueras är att generera ett surrogat som konvergerar mot ett eftersökt resultat, samtidigt som det behöver så få simuleringsbedömningar som möjligt.



Figur 2.5: Flödesschema för surrogatmodellering.

I figur 2.5 observeras ett flödesschema för surrogatmodellering [13]. Surrogatprocessen innefattar de tre huvudstegen samplingsintervall, konstruktion av surrogatmodellen och bedömning av noggrannheten hos surrogatet. Nedan beskrivs de modeller som används i respektive huvudsteg.

2.4.1.1 Latin hypercube sampling

Ett sätt att definiera punkter i Lösningsrymden är genom en modell som kallas *Latin hypercube sampling*, förkortat LHS [13]. En LHS ger randomiserade punkter i en kub, där kuben representerar Lösningsrymden och punkterna värden på optimeringsvariablerna. Punkterna placeras automatiskt på ett sådant sätt att indatan sprids ut tillräckligt mycket så att inga eventuella optimala värden går förlorade samt att variablerna inte begränsas av en för liten Lösningsrymd. Ett önskat antal noder och variabler ger dimensionen på Lösningsrymden. Metoden används sedan för att maximera det minimala avståndet mellan varje nod och därmed täcker Lösningsrymden ett så stort område som möjligt inom de givna intervallen för variablerna. Vidare observeras den aktuella LHS som används under rubrik 3.2.3 samt i figur 3.9.

2.4.1.2 Radial basis function

Surrogatmodellering är ett sätt att utföra traditionell optimering, även kallat *design of experiments*, förkortat DOE. För optimering med surrogatmodellen används interpolationsmetoder. Interpolering innebär att en funktion utgår från ett antal kända datapunkter och därefter beräknar nya värden i okända punkter. En välanvänd interpolationsmetod kallad *radial basis function*, förkortat RBF, går ut på att beräkna det euklida avståndet från den sökta punkten hos funktionsvärdet till varje given nod [14]. Den generella formeln för RBF ges av

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T \mathbf{\Psi} = \sum_{i=1}^n w_i \psi(r) = \sum_{i=1}^n w_i \psi(\|\mathbf{x} - \mathbf{x}_i\|) \quad (2.9)$$

där $\psi(r)$ ansätts beroende på vilken typ av RBF som är önskvärd för något $r \geq 0$, och $r = \|\mathbf{x} - \mathbf{x}_i\|$. Variabeln w är en viktad funktion beroende av $\psi(r)$. Fördelen med denna interpolationsmetod är att RBF inte blir mer komplicerad om antalet optimeringsvariabler önskas öka. Ekvation (2.9) leder till det linjära symmetriska systemet

$$\mathbf{y} = \mathbf{\Upsilon} \mathbf{w} \quad (2.10)$$

där $\mathbf{\Upsilon}$ är definieras som $v_{i,j} = \psi(\|\mathbf{x}_j - \mathbf{x}_i\|)$, $i, j = 1, 2, \dots, n$, och betecknar den så kallade *Gram matrisen* [13]. Vilken typ av RBF, $\psi(r)$, som bestäms ansätts slutligen och tabell 2.1 presenterar de vanligaste typerna [15].

Tabell 2.1: Typ av RBF.

Typ	Funktion
Gaussisk	$\psi(r) = e^{-(\epsilon r)^2}$
Multikvadratisk	$\psi(r) = \sqrt{1 + (\epsilon r)^2}$
Inverterad kvadratisk	$\psi(r) = \frac{1}{(\epsilon r)^2}$
Inverterad multikvadratisk	$\psi(r) = \frac{1}{\sqrt{1 + (\epsilon r)^2}}$
"Spline"-funktion	$\psi(r) = r^k, k = 1, 3, 5 \dots$
"Spline"-funktion	$\psi(r) = r^k \ln(r), k = 2, 4, 6 \dots$

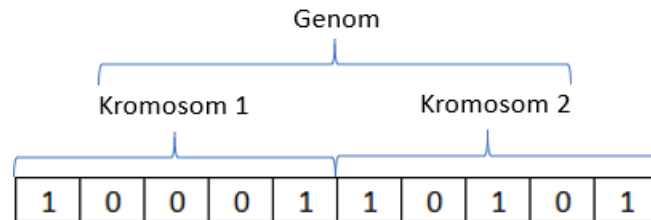
2.4.1.3 Bedömning av resultat

Surrogatets noggrannhet beror av antalet iterationer och positioner i lösningsrymden. Eftersom optimeringen utförs med hjälp av en interpolationsmetod, se avsnitt 2.4.1.2, bedöms resultatet från RBF nått det bästa värdet då resultatet av optimeringsvariablerna konvergerat mot ett värde för respektive variabel. Iterationer utförs därför kontinuerligt tills detta är uppfyllt.

2.4.2 Genetiska algoritmer

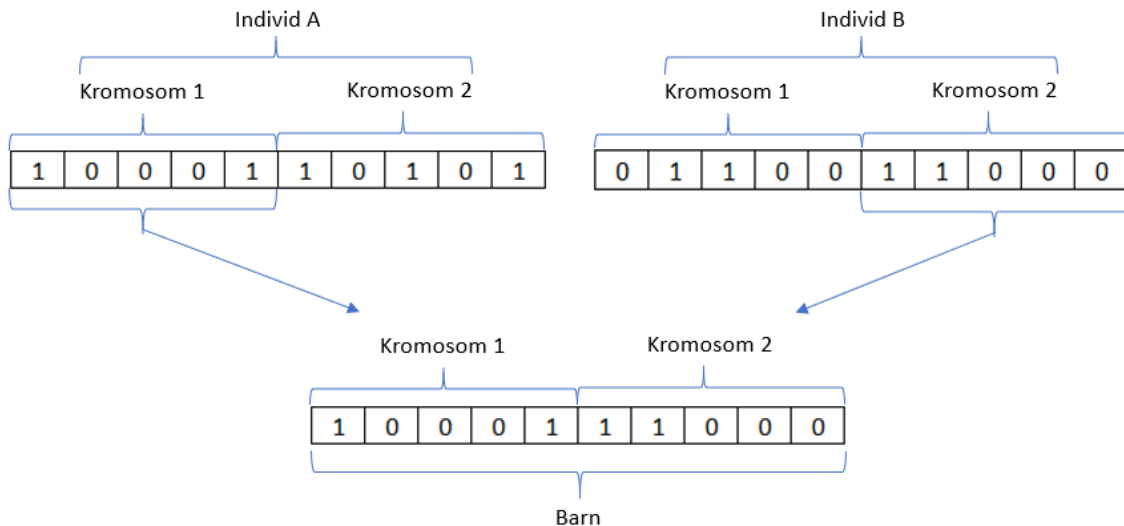
Precis som namnet antyder grundar sig en genetisk algoritm i att efterlikna Darwins evolutionsteori om naturligt urval. Metoden extruderas alltså från biologin och biologins termer används. Vissa förenklingar av verklig evolution görs. Algoritmen kan modifieras på flera sätt för olika funktioner. Nedan följer en övergripande beskrivning av en generell metod [16].

Till en början skapas en population av randomiserade individer. Denna population är första generationen, och kommande generationer härstammar från dessa individer. En individs genom består, till skillnad från verkligt DNA, oftast av binär kod. Ett alternativ är att spara värdena direkt utan att omvandla dem till binär kod. Genomet delas upp i två lika stora kromosomer, se figur 2.6, vilka senare används vid skapandet av nästa generation.



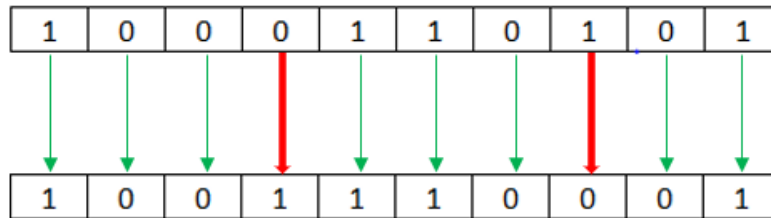
Figur 2.6: Uppbyggnad av en individs genom.

Individerna testas nu för att se vilka som är bäst anpassade för den givna uppgiften, och rankas utefter resultatet. Denna ranking står till grund för vilka individer som har störst chans för reproduktion. Individer är androgyna, då alla individer kan få barn med varandra. Barnen skapas genom att halva genomet kommer från två slumpartade individer, där individerna bidrar med var sin kromosom vilket visas i figur 2.7. De individer som rankats högt har alltså större chans att sprida sitt genom vidare till flera barn. En vanlig metod att implementera är *elitism*, vilket innebär att den bästa individen alltid överlever och kopieras till nästa generation. Detta är för att utesluta att den bästa individen inte ska försvinna till efterföljande generation, och har som konsekvens att algoritmen kan konvergera snabbare.



Figur 2.7: Tillvägagång för att skapa ett barn från två individer.

Nästa steg är att mutera avkomman. Varje barns genom kan här slumpvis förändras. Detta görs genom att varje binär bit av genomet har en liten chans att ändra värde, se figur 2.8. Mutationschansen ligger ofta mellan 2-5 % per bit. Efter mutation är de mogna att testas. Barnen är nu nästa generation och kommer utsättas för samma prövning som sina föregångare.



Figur 2.8: En individ före och efter mutation.

Generellt fungerar metoden bättre med stora populationer och ett högt antal generationer. Att maximera antalet generationer gånger antalet individer är fördelaktigt. Exakt hur stora populationer och hur många generationer som ska finnas i en genetisk algoritm skiljer sig beroende på problemet som ska lösas men det är rekommenderat att ha åtminstone 30 individer per generation och 100 generationer. Det är dock möjligt att med vissa modifikationer ha en populationsstorlek om endast en individ och förlita sig helt på mutationer.

2.4.3 Adjointmetoden

Adjointmetoden är en gradientbaserad formoptimeringsmetod som utvecklades redan under 1970-talet. Inledningsvis var adjointmetoden ett koncept inom statiken och dynamiken men har under de senaste 10-20 åren blivit en erkänd metod i strömningmekaniken [17]. Den övergripande idén med adjointmetoden är att utifrån en CFD-simulering se hur geometrin för en viss kropp ska optimeras. Detta görs utifrån så kallad känslighetsdata som erhålls från CFD-simuleringar.

Adjointmetoden är en effektiv metod inom CFD då det endast krävs två lösningssteg, en primärlösning och en adjointlösning, i varje optimeringscykel. Metoden är oberoende av antalet designvariabler problemet innefattar, vilket innebär att beräkningar kan utföras relativt snabbt [17]. Primärlösningen som erhålls efter flödessimuleringen ger ett lösningsfält. Efter lösning av de så kallade adjointekvationerna med lösningsfältet som indata erhålls adjointlösningen efter att en kostnadsfunktion har definierats. Kontrollpunkter definieras vid geometrin för att den senare ska kunna deformeras. Ytkänslighetsdata beräknas för den specifika geometrin och visualiseras sedan på ytan till vingen. Detta ger så kallade känslighetskartor [17]. Dessa kartor ger en intuitiv bild av vilka förändringar som behövs för att optimera kostnadsfunktionen och var dessa förändringar bör ligga.

Utifrån känslighetsdatan som erhålls från adjointlösaren deformeras sedan meshen. En ny optimeringscykel påbörjas sedan genom körning av flödeslösaren och adjointlösaren som ger ny känslighetsdata för den förändrade geometrin.

Flera tidigare projekt [18, 19] har visat att gradientbaserade optimeringsmetoder som adjointmetoden är effektiva när det gäller formoptimering av vingprofiler. Metoderna är välkända för att kunna ge aerodynamiska förbättringar efter relativt få cykler.

2.4.3.1 Adjointekvationen

En av de viktigaste ekvationerna för adjointmetoden är adjointekvationen

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}}\right)^T \cdot \Lambda = \frac{\partial J}{\partial \mathbf{q}} \quad (2.11)$$

där \mathbf{R} är residualerna från Navier-Stokes, \mathbf{q} flödeslösningen, Λ det adjointa fältet och J kostnadsfunktionen som adjointlösaren optimerar mot. $\left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}}\right)^T$ visar på känsligheten av flödesresidualerna avseende flödeslösningen och $\frac{\partial J}{\partial \mathbf{q}}$ är derivatan av kostnadsfunktionen med avseende på flödeslösningen. Lösningen av det adjointa fältet Λ som i sin tur ger adjointlösningen fås genom lösning av ekvationssystemet (2.11).

3

Metod

I detta kapitel beskrivs den metodik som projektet använde sig av. För att kunna applicera de tre optimeringsstrategierna surrogatmodellering, genetiska algoritmer, och adjointmetoden utfördes en del förarbete. Förarbetet omfattade konstruerandet av en funktionell testmiljö att utgå från varpå en beräkningsdomän kunde byggas där sedan en mesh skapades. Därefter beskrivs hur de tre optimeringsstrategierna applicerades på detta projekt.

3.1 Förarbete

Innan de tre optimeringsmetoderna applicerades behövdes en funktionell testmiljö. Då inga fysiska experiment gjordes användes istället STAR-CCM+ för att simulera luftflödet. I STAR-CCM+ ställdes en beräkningsdomän upp med en fysisk modell för luftflödet. I beräkningsdomänen genererades ett mesh vilket lösningen beräknades på.

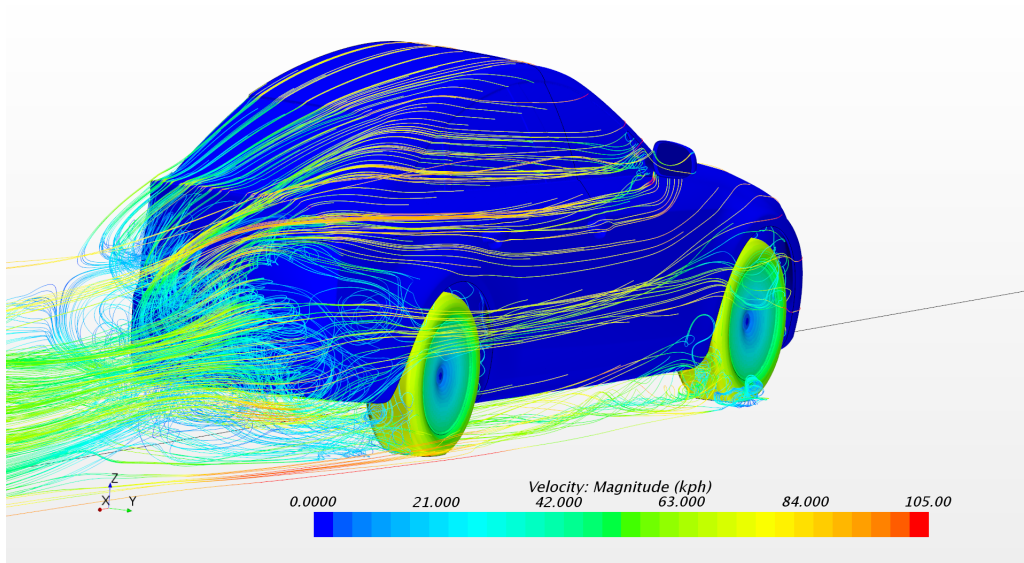
3.1.1 DrivAer-modellen utan bakvinge

För att bedöma hur de olika metodernas resultat ställer sig mot varandra användes en referens bestående av DrivAer-modellen utan bakvinge. Referensen användes även för att bedöma hur mycket negativ lyftkraft bakvingen tillförde bilen samt hur mycket luftmotståndet ökade.

Tabell 3.1: Resultat för DrivAer-modellen utan bakvinge.

C_L -värde	C_D -värde
0,0049	0,2291

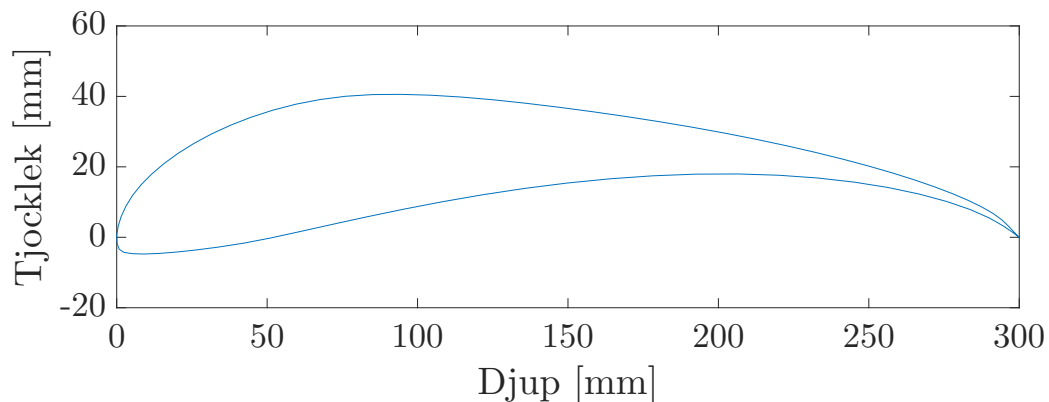
Efter simulering av DrivAer-modellen utan bakvinge observerades luftmotståndet C_D och lyftkraften C_L , se tabell 3.1. Simuleringen gav även en bild av hur luften strömmade kring bilen, se figur 3.1. Hur luften strömmade över bilen är väsentlig information då attackvinkeln mättes gentemot luftströmmen.



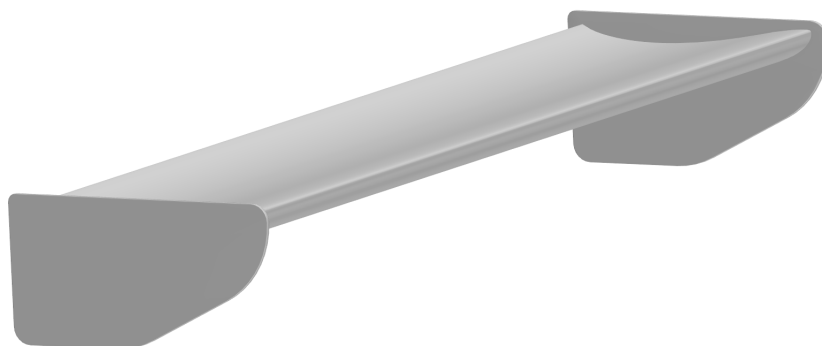
Figur 3.1: Strömlinjer för DrivAer-modellen utan vinge.

3.1.2 Konstruktion av bakvinge

För att kunna genomföra en aerodynamisk formoptimering av en bakvinge konstruerades en bakvinge utifrån en NACA Airfoil med beteckningen S1223 [20], figur 3.2. Denna roterades sedan 180 grader för att uppnå negativ lyftkraft istället för lyftkraft. Utifrån mått på vingar på GT3-bilar bestämdes lämpliga dimensioner på vingen, se appendix A, figur A.1. På denna vinge placerades sedan sidopaneler för att hjälpa till att styra luftflödet. Dessa visas också i figur A.1. Sidopanelerna ansågs vara en del av vingen och vreds därför med vingprofilen när attackvinkel ändrades. En begränsning för denna vinge jämfört med en verklig vinge är att denna inte har några stöd. Istället hängde den i luften bakom bilen, se figur 3.3.



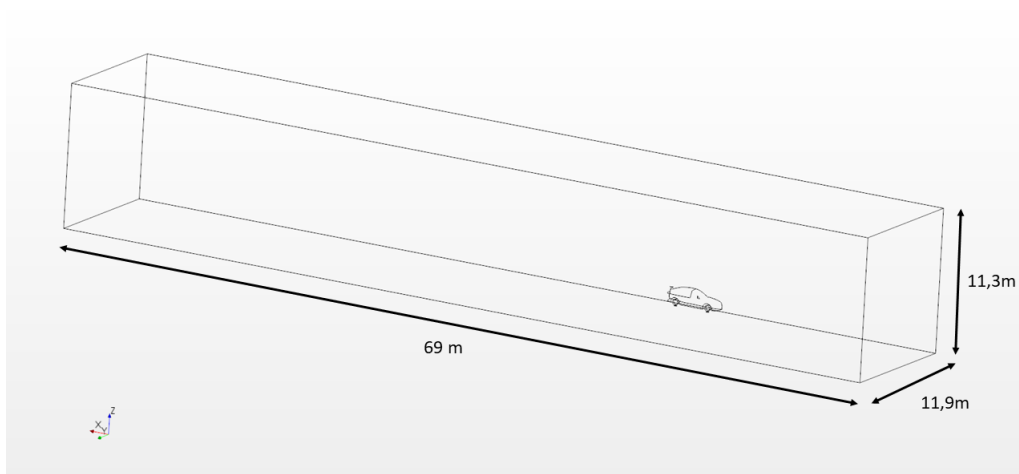
Figur 3.2: NACA-profil, S1223 - Selig S1223 high lift low Reynolds number airfoil [20].



Figur 3.3: Den konstruerade vingen för användning till optimeringsmetoderna.

3.1.3 Beräkningsdomän

En beräkningsdomän konstruerades med mått enligt figur 3.4. Luften strömmade in via inloppet från den högra kortsidan i figuren och träffade DrivAer-bilen rakt framifrån. Viktigt att notera är att detta är en domän för halva bilen, då den har delats längs bilens symmetriplan. Detta för att flödet kring bilen antogs vara symmetriskt vilket medför att beräkningstiden minskar. Randvillkoren visas i tabell 3.2.



Figur 3.4: Dimensioner för beräkningsdomänen. Notera att domänen endast inkluderade halva bilen, då den är delad längs bilens symmetriplan.

Tabell 3.2: Randvillkor för beräkningsdomän med DrivAer-bil.

Rand	Randvillkor
DrivAer	<i>Wall</i>
Inlopp	<i>Velocity inlet</i>
Utlopp	<i>Pressure outlet</i>
Golv	<i>Moving wall, 100 kph</i>
Tak	<i>Symmetry plane</i>
Symmetriplan	<i>Symmetry plane</i>
Sida	<i>Symmetry plane</i>
Däck	<i>Rotating wall</i>

3.1.4 Fysikalisk modell

Valet av den fysikaliska modellen visas i tabell 3.3. Luftströmmens hastighet valdes till 100 km/h med en temperatur på 20 °C. Reynoldstalet blev därmed ungefär $8,16 \cdot 10^6$ och ansågs vara tillräckligt stort för att säkerställa att C_D -värdet var svagt beroende av Reynoldstalet [2]. Vid dessa värden kan densiteten antas vara konstant. Vidare användes RANS och steady state för att förenkla beräkningarna och därmed reducera beräkningstiden. $k - \omega$ modellen användes av anledningar diskuterade i kapitel 2.1.2.

Tabell 3.3: Fysikaliska modellen för CFD.

Fysikaliska Modeller
<i>All y^+ Wall Treatment</i>
<i>Constant Density</i>
<i>Coupled Flow</i>
<i>Exact Wall Distance</i>
<i>Gas \rightarrow Air</i>
<i>Gradients</i>
<i>k-omega Turbulence</i>
<i>Reynolds-Average Navier-Stokes</i>
<i>SST (Menter) k-omega</i>
<i>Steady</i>
<i>Three Dimensional</i>
<i>Turbulent</i>

3.1.5 Solver-inställningar

För att lösaren skulle konvergera snabbare och med högre precision ändrades inställningar i lösaren. De finns presenterade i tabell 3.4. En ökning av Courant numret ger en snabbare konvergens dock med nackdelen att lösaren blir känsligare för fel och har större risk för att divergera.

Tabell 3.4: Solverinställningar i STAR-CCM+.

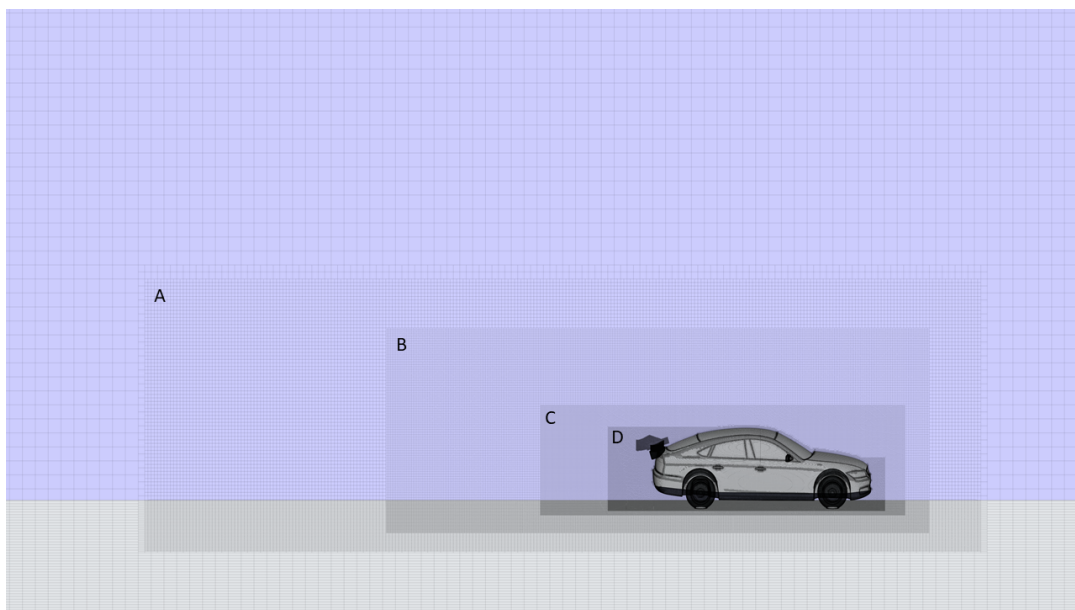
Inställning	Värde
<i>Courant number</i>	50
<i>normalized Flat Cells Curvature Factor</i>	1E-20
<i>Expert Drivers</i>	On
<i>AMG Linear Solver → V Cycle → Pre Sweeps</i>	2
<i>AMG Linear Solver → V Cycle → Post Sweeps</i>	3

3.1.6 Mesh

För att lösa beräkningarna delades volymen upp i ett ändligt antal celler. Cellerna som användes är av typen *Trimmed Cells*, vilket är hexahedrala celler. Fördelen med att använda denna typ av celler är att de kan växa väldigt fort från små till stora celler. Beroende på komplexitet i geometrin och flödet krävs olika stora celler i olika delar av meshet. Närmst ytmeshet genererades det ett *Prism Layer* som är bra för att följa kurvatur på ytan och som kan växa gradvis från ytan för att vara tillräckligt små närmast ytan. Detta för att få ett bra y^+ , se rubrik 2.2.

3.1.6.1 Meshgenerering

Då flödet antogs vara likt friström långt bort från bilen krävdes inte små celler där. Närmre bilen behöver cellerna vara mindre och därför lades det till *Density Boxes* där cellstorleken förfinades, vilka visas i figur 3.5 samt tabell 3.6. Då bilen störde flödet behövde cellerna vara finare en längre sträcka bakom bilen än framför.



Figur 3.5: Förfining av volymmesh genom stegvis minskande *Density Boxes* runt bilen, med en minskad cellstorlek i varje. Inställningar för dessa presenteras i tabell 3.6.

Inställningarna för meshgenereringen visas i tabell 3.5. En *Base Size* valdes till 8,5 millimeter som inställningar *Relative to Base* och *Percentage of Base* är relativa till. Ett $y^+ < 1$ eftertraktades på vingen, och därför lades det till en *Surface Control* på vingen. En *Surface Control* görs för att ändra inställningarna för ytmeshen i ett visst område. Flödet runt vingen är intressant att studera i detalj och därför behöver cellerna kunna lösa upp volymen ordentligt. Dessa inställningar visas i tabell 3.7. Det gjordes även en *Surface Control* på beräkningdomänens alla väggar där *Target Surface Size* sattes till *Relative to Base* - 3200 % för att få stora celler i friströmmen. Totalt genererades ungefär 31 miljoner celler.

Tabell 3.5: Inställningar för den genererade meshen.

Parameter	Inställning	Värde
<i>Base Size</i>	<i>Absolute Value</i>	0,0085 m
<i>Target Surface Size</i>	<i>Relative to Base</i>	100 %
<i>Minimum Surface Size</i>	<i>Relative to Base</i>	25 %
<i>Surface Curvature</i>	<i># pts/circle</i>	360
<i>Surface Growth Rate</i>	-	1,2117
<i>Number of Prism Layers</i>	-	8
<i>Prism Layer Near Wall Thickness</i>	-	0,0018 m
<i>Prism Layer Total Thickness</i>	<i>Absolute</i>	0,031387 m

Tabell 3.6: Inställningar för *Density Boxes* som är illustrerade i figur 3.5.

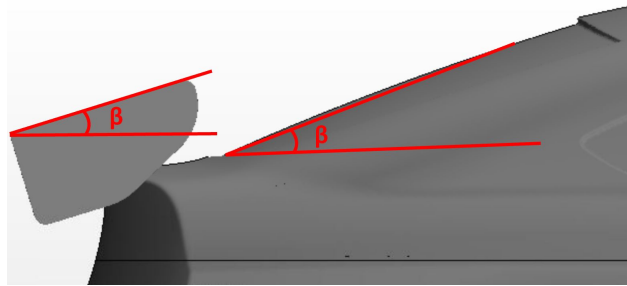
Parameter	Inställning	Värde	Längd [m]	Höjd [m]
A	<i>Relative to Base</i>	800 %	17,78	4,32
B	<i>Relative to Base</i>	400 %	11,40	3,32
C	<i>Relative to Base</i>	200 %	7,68	1,88
D	<i>Relative to Base</i>	100 %	5,84	1,40

Tabell 3.7: Inställningar för *Surface Control* på vingens yta.

Parameter	Inställning	Värde
<i>Target Surface Size</i>	<i>Relative to Base</i>	25 %
<i>Minimum Surface Size</i>	<i>Relative to Base</i>	12.5 %
<i>Number of Prism Layers</i>	-	16
<i>Prism Layer Near Wall Thickness</i>	-	1,217411 E-5 m
<i>Prism Layer Total Thickness</i>	<i>Absolute</i>	0,0055 m
<i>Wake Refinement</i> → <i>Isotropic Size</i>	<i>Percentage of Base</i>	25 %

3.2 Optimering

De variabler som till en början optimerades inom surrogatmodellen och den genetiska algoritmen var höjd från bakluckan på bilen och attackvinkel. Anledningen till att endast två variabler användes från start var för att få någon form av grund att utgå ifrån, samt att ingen vetskap fanns om hur tidskrävande det kommer vara för en variabel att konvergera. Attackvinkeln mättes gentemot luftströmmen som är parallell med bakrutan, se figur 3.1. Vingen har alltså en attackvinkel på noll grader när vingen är i samma vinkel som bilens bakruta vilket illustreras i figur 3.6.



Figur 3.6: Attackvinkel noll grader för vingen sker när vingen och bakrutan har samma vinkel.

För surrogatmodellen adderades ytterligare en variabel då det observerades att de första två variablerna konvergerade mot varsitt värde relativt snabbt. Därav ökades möjligheten till att förbättra lyft- och dragkrafterna på bilen ytterligare. Variabeln som adderades var position för vingen ovanför backluckan relativt ett lokalt koordinatsystem. Vingen kunde då också flyttas i longitudinellt led, se figur 3.7.

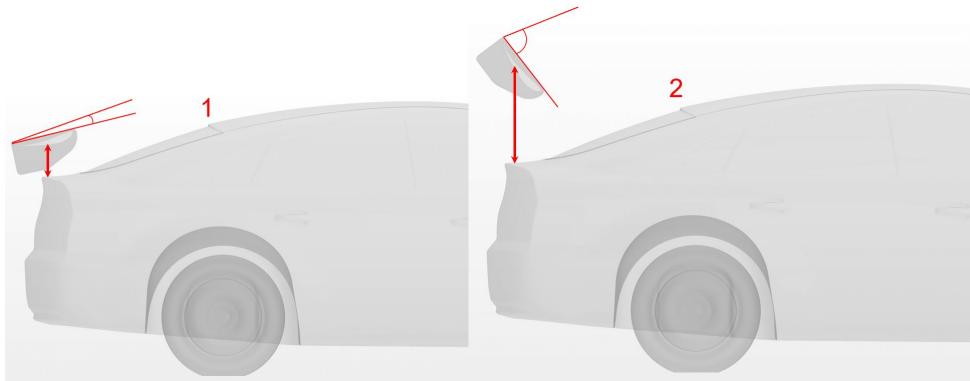


Figur 3.7: Vinge relativt lokala koordinatsystemet på bakluckan.

Lösningensrymdens storlek bestämdes genom att välja mellan vilka intervall punkterna skulle ligga. Intervallet 0,2 till 0,5 meter ansattes för höjden samtidigt som attackvinkeln ansattes till intervallet 5 till 70 grader. Intervallet för attackvinkeln skulle täcka in väldigt aggressiva attackvinklar samt väldigt milda attackvinklar vilket intervallet 5 till 70 grader innefattar. Maximala höjden valdes till 0,5 meter för att lösningensrymden skulle bli stor även gällande höjdvariabeln. Nackdelen med att välja så stor maximal attackvinkel blev att höjder under 0,2 meter inte kan testas då en aggressiv vinkel tillsammans med en låg höjd gör att vingen skär in i bilen. Den optimala höjd- och vinkelkombinationen antogs dock ligga inom dessa intervall.

Skulle optimeringsmetoderna generera en optimal höjd på 0,2 meter och inte en så aggressiv attackvinkel fortsattes studien genom att ändra intervallen så att mindre höjder också täcks in.

Figur 3.8 tydliggör intervallet för höjd och vinkel på vingen. I figurens första bild, till vänster, är höjden 0,2 meter och attackvinkeln 5 grader. Figurens andra bild, till höger, innehar en attackvinkel på 70 grader och en höjd på 0,5 meter. Det är alltså inom dessa intervall som olika höjder och vinklar simulerades i STAR-CCM+.

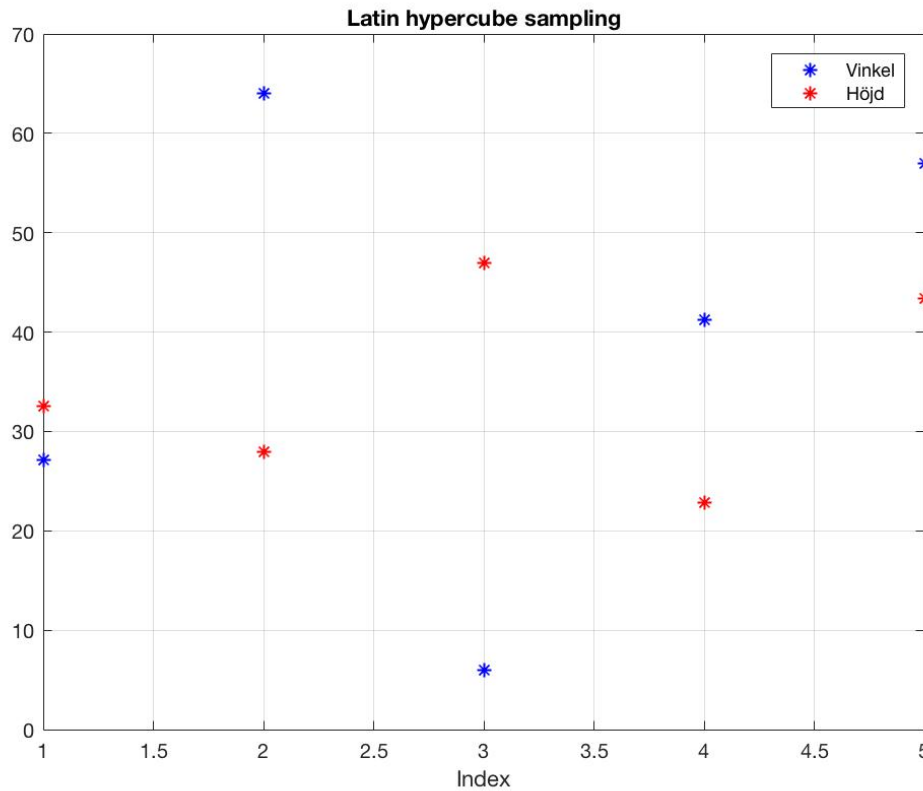


Figur 3.8: Intervall för höjd och vinkel. Denna figur visar också tydligt att vingens sidopaneler följer med vid vinkeländring av vingprofilen.

Adjointmetoden skiljer sig i sättet metoden utförs på eftersom den har svårare att ta hänsyn till olika optimeringsvariabler på samma sätt som surrogatmodellen och den genetiska algoritmen. På grund av detta formoptimerades istället geometrin vid tillämpningen av adjointmetoden. I detta arbete utfördes tillämpningen med hjälp av STAR-CCM+.

3.2.1 Latin hypercube sampling

Processen för genomförande av surrogatmodellen och den genetiska algoritmen inleddes med att generera ett godtyckligt antal värden i lösningsrymden. Antalet startpunkter ansattes till fem stycken och med hjälp av Latin hypercube sampling spreds dessa ut i lösningsrymden. Varje startpunkt skulle innehålla en godtycklig höjd över bakluckan på bilen och en godtycklig attackvinkel mot luftflödet sett från bilens bakruta.



Figur 3.9: Latin hypercube sampling. Notera att enheten för höjd är i centimeter.

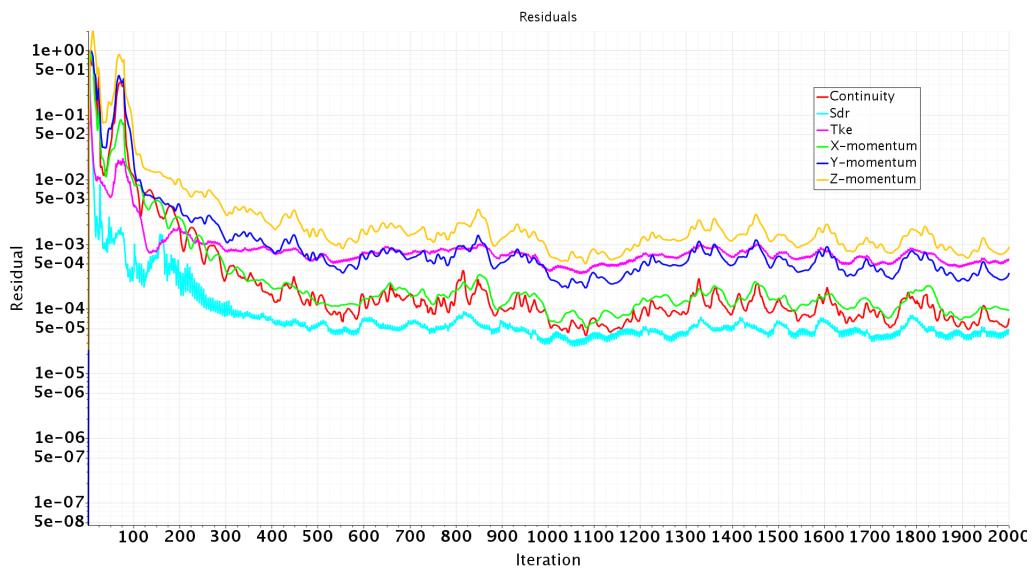
För att skapa en LHS i lösningsrymden användes MatLab och en modifierad version av dess inbyggda funktion *lhsdesign.m*. Indata för att skapa en LHS är antalet variabler som skall användas, vilket i denna studie till en början var attackvinkel och höjd. Antalet noder som ska användas är också indata, vilka valdes till fem stycken per variabel. Intervallen för de två variablerna definierades, se rubrik 3.2. Utdata blev i form av en plot där x-axel anger index. Varje index har två tillhörande noder där den ena är attackvinkel och den andra är höjden över bilens baklucka. På y-axeln angavs höjden i centimeter och attackvinkel i grader. Alla noder i LHS observeras i figur 3.9 och exakta värden för variablerna observeras i tabell 3.8.

Tabell 3.8: Värden från LHS.

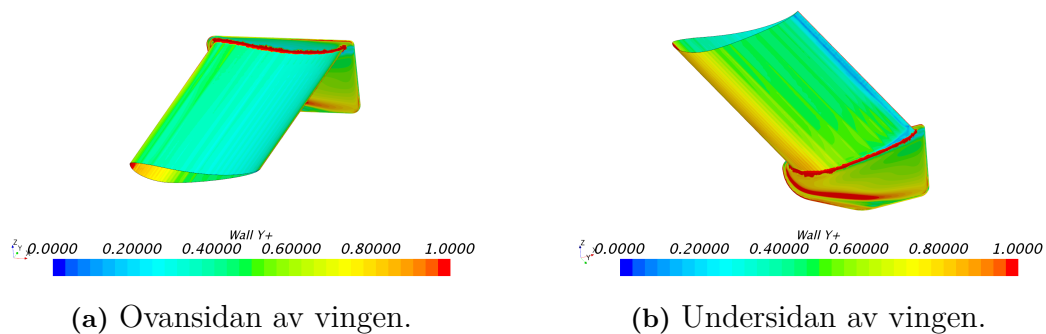
Höjd [m]	Attackvinkel [grader]
0,3256	27,15
0,2792	63,98
0,4691	5,97
0,2287	41,19
0,4338	56,99

3.2.2 Konvergens

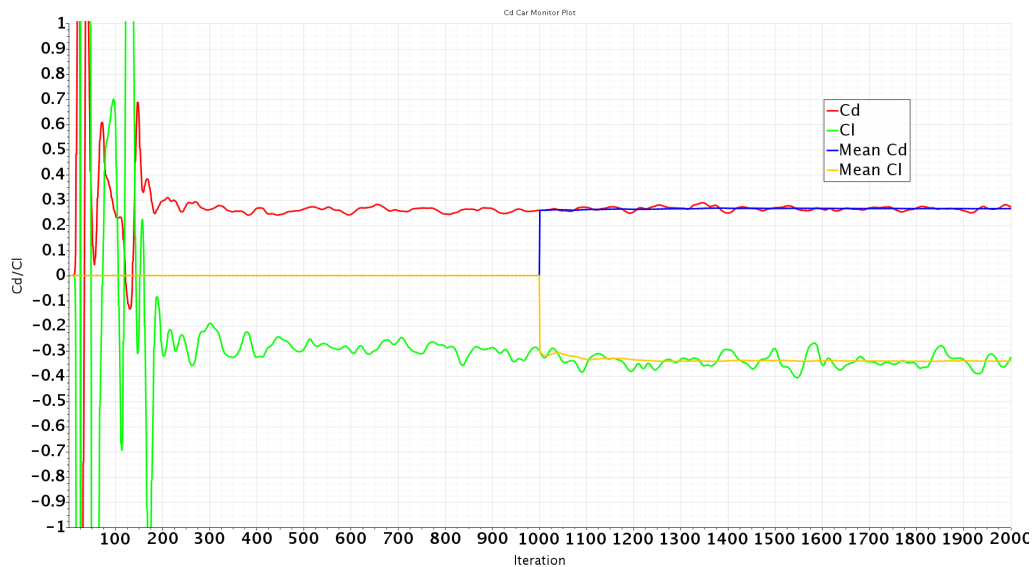
För optimering med DOE metoder är huvudkravet att lösningarna uppnår konvergens. För detta kandidatarbete betyder det att residualerna för lösningen konvergerat mot värden runt 10^{-4} . Idealt är det önskvärt att residualerna konvergerar mot maskinprecision, men då geometrin var så pass komplex kunde detta aldrig uppnås. För det generella fallet med CFD beräkningar anses residualer med ett värde $\leq 10^{-4}$ ha konvergerat [21]. Residualerna observerades efter färdig simulering i STAR-CCM+ och visas i figur 3.10. Vidare kontrollerades att y^+ på bilen konvergerat mot ett värde på cirka 60 och y^+ på vingen antar värden under ett, se figur 3.11 samt se rubrik 2.2 för vidare information kring y^+ . Medelvärden för C_D och C_L kontrollerades slutligen så att det inte fluktuerade mellan olika värden utan höll ett konstant värde mellan den tusende och tvåtusende iterationen, se figur 3.12. Viss fluktuation accepterades för det absoluta värdet på C_D och C_L så länge den var regelbunden.



Figur 3.10: Residualer för en simulering ligger mellan $5 \cdot 10^{-3}$ och $5 \cdot 10^{-5}$.



Figur 3.11: Visualisering av y^+ kring vingen.



Figur 3.12: Absoluta värdet samt medelvärdet för C_D och C_L . C_D värdet har konvergerat bra, medan C_L fluktuerar. Dock är fluktuationen tillräckligt jämn och därför anses medelvärdet ha konvergerat.

3.2.3 Surrogatmodellering

Processen för genomförande av surrogatmodellering inleddes med att använda de startvärden som genererats av LHS, se tabell 3.8. Samplingsplanen från LHS gav de fem första konfigurationerna som simulerades i STAR-CCM+. En ny höjd och vinkel ersatte en utgångshöjd på 0,1025 meter från bakluckan och en attackvinkel på noll grader mot bakrutan. Simuleringarna gjordes på Chalmersklustret Vera och varje simulering tog cirka 250 core-timmar. När simuleringen var färdig noterades det om lösningen hade konvergerat och om ett medelvärde av C_L och C_D kunnat beräknas. Kraven för att en lösning skall anses ha konvergerat observeras under rubrik 3.2.2.

Nya optimeringsvariabler genererades med hjälp av ett generellt surrogatskript skrivet i programmet Julia av Magnus Urquhard, doktorand vid avdelningen för tillämpad mekanik, Chalmers tekniska högskola. Handledare Adam Brandt, doktorand vid samma avdelning, skrev sedan ett mindre skript som använde sig av Urquhards generella skript för att anpassa genererade värden till detta kandidatarbete. Tillammans gav dessa skript två nya värden på höjd och vinkel, en standardavvikelse, benämnt STD, och ett median-värde. Medianen är en punkt i lösningsrymden där programmet förutsåg att ett bättre värde på C_D skulle genereras med bibehållet C_L . Standardavvikelsen är en punkt i lösningsrymden där programmet önskade få mer information om rådande förhållanden för att kunna ge ett bättre värde på nästkommande median. Om surrogatmodellen gav ett värde av randomiserad typ, benämnt RAND, betyder det att det är en simulering med randomiserade värden. Ett randomiserat värde på optimeringsvariablerna antogs uppkomma då surrogatskripten inte kunde hitta en bättre median än vad som redan hade simulerats. En ny simulering skapades därefter i STAR-CCM+ med de nya värdena för höjd och vinkel och därmed fortsatte denna arbetsgång fram till att Julia-skripten återkommande gånger

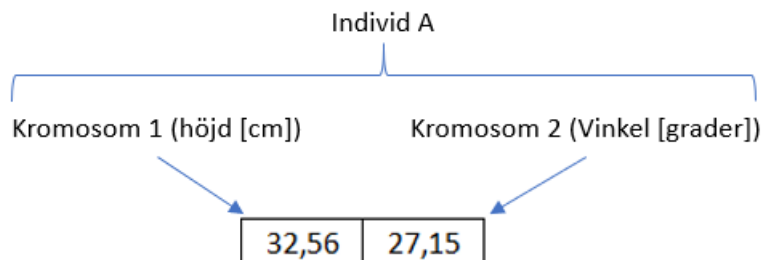
föreslog ett median-värde som var väldigt likt det tidigare värdet. Optimeringen antogs då vara klar och variablerna hade konvergerat mot en höjd och en vinkel. Vid detta tillfälle lades variabeln longitudinellt led till som optimeringsvariabel.

3.2.4 Genetiska algoritmer

En genetisk algoritm skrevs i MatLab och grunden för hur denna är uppbyggd hittas under 2.4.2. I följande avsnitt förklaras vilka variabler som valdes samt anledningar till dessa.

Först bestämdes antal individer och generationer. Detta var ett viktigt val då fler individer och generationer skulle innebära längre beräkningstid. Totala antalet simuleringar begränsades till 25 stycken. Denna begränsning innebar att fem individer kunde skapas samt minst fem generationer. För att minska antalet simuleringar var det viktigt att identiska individer inom och mellan generationer inte simulerades flera gånger. Därför sattes en gräns om att individer som skiljer sig mindre än $\pm 0,005$ på samtliga variabler inte skulle simuleras igen.

Den första generationen skapades med hjälp av LHS, samma variabler som användes för surrogatmodellens första simuleringar användes här som första generationens individer. Istället för att beskriva genomet med binär kod användes variablernas direkta värde vilket visas i figur 3.13.



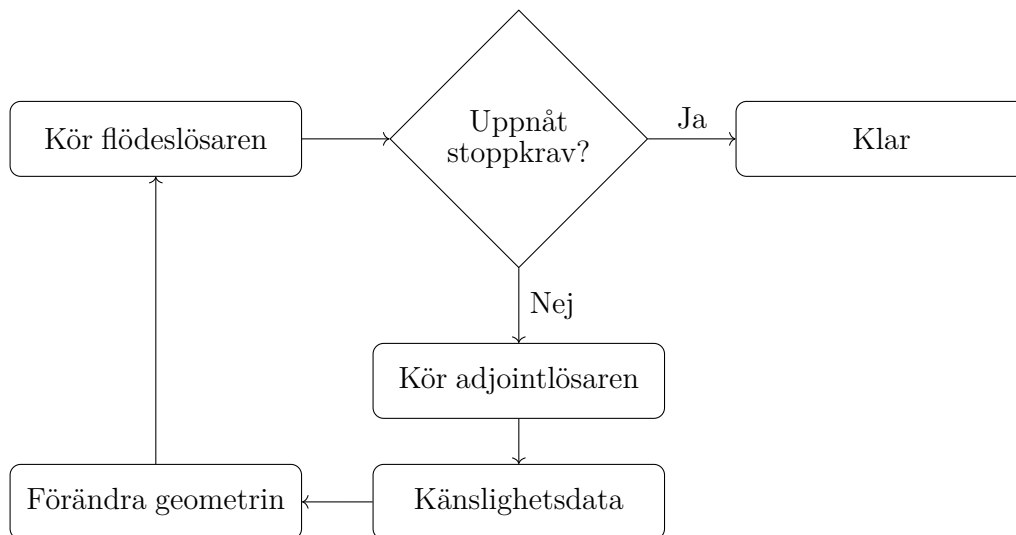
Figur 3.13: Uppbyggnad av verkligt genom för en individ.

Individerna testades genom att simulera varje unik individ. Individerna rankades sedan efter hur bra C_D -värde och C_L -värde de hade relativt de andra individerna, där individen bestraffades hårt för ett C_L -värde över $-0,35$. En högt rankad individ hade då störst chans att föra sina gener vidare gentemot en lågt rankad individ.

Nästa generation skapades genom korsbefruktning. De nya barnen muterades och eftersom variablerna inte är binära muterades de istället över respektive intervall de var definierade för med hjälp av Matlabs randomiserade tal. Detta skiljer sig något från teorin vilken beskrivs under kapitel 2.4.2. Syftet är dock att genomet ska ändras slumpartat vilket det har. De första generationerna hade en stor chans att mutationen skedde till vilken punkt över intervallen som helst och en liten chans att det endast ändrades lite. En liten ändring kallas *creeping*. För varje generation blev mutation över hela intervallet mer sällsynt medan chansen för små mutationer ökade. Totala chansen för mutation hölls konstant på 20 % för varje kromosom.

3.2.5 Adjointmetoden

Arbetsprocessen för adjointmetoden presenteras i figur 3.14. Flödeslösaren körs först för att sedan kontrollera om stoppkravet uppnåtts. Kravet kan vara ett visst antal iterationer eller att lösaren har konvergerat. Därefter körs adjointlösaren baserat på vad som skall optimeras. Adjointlösningen erhålls vilket innebär att ytkänslighetsdatan kan beräknas. Från denna kunde sedan geometrin förändras i STAR-CCM+ genom utplacering av kontrollpunkter och en ny flödeslösning beräknas. Metoden bygger på att göra små förändringar i varje cykel och därefter köra en flödessimulering för den nya geometrin. Varje cykel resulterade därför i en liten förbättring. Efter ett antal cykler kan det totalt sett uppnås en stor förbättring jämfört med den ursprungliga geometrin. I denna studie genomfördes bara beräkningar fram till och med känslighetsdatan.



Figur 3.14: Arbetsprocessen vid användning av adjointmetoden.

Normalt kan inte den inbyggda adjointlösaren i STAR-CCM+ optimera för höjd och attackvinkel, som var variablerna som användes i surrogatmodellen och i den genetiska algoritmen. För att optimera enligt dessa krävs externa skript. Av denna anledning användes adjointmetoden genom att förändra vingens form istället för att optimera höjd och attackvinkel.

Vid implementeringen av adjointmetoden på DrivAer-modellen användes den första av de tidigare simuleringarna från Latin hypercube sampling, se kapitel 3.2 och första raden i tabell 3.8. Utgångsvärdena på C_D och C_L skiljer sig något från de för simulering 1 i tabell B.1, eftersom dessa är medelvärderade och adjointlösaren kan inte använda kostnadsfunktioner som baseras på medelvärderade värden. Därför användes de faktiska värdena på C_D och C_L vid flödessimuleringens slut som utgångsvärden.

3.2.5.1 Kostnadsfunktion

För att kunna minimera luftmotståndet och samtidigt behålla den negativa lyftkraften från referensfallet användes två egna kostnadsfunktioner. Detta eftersom optimering med de inbyggda funktionerna oftast leder till att när luftmotståndet minskar så minskar även den negativa lyftkraften. Kostnadsfunktionen skapades utifrån en så kallad *Expression Report* i STAR-CCM+. Idén var att efterlikna en stegfunktion genom att undersöka om värdet på lyftkraften, C_L , gick över ett visst straffvärde och då antog funktionen ett högt värde. På detta sätt straffades lösningar som gav ett för högt värde på C_L . Om C_L inte övergick straffvärdet antog den skapade rapporten värdet för luftmotståndet, C_D .

En funktion som testades i en *Expression Report* för olika fristående testvingar, dock inte den som presenteras i kapitel 3.1.2, var en stegfunktion, i form av en if-sats, som antog värdet för luftmotståndet när den negativa lyftkraften gick under ett visst straffvärde. Värdet valdes till lite större än det ursprungliga värdet på den negativa lyftkraften. Om den negativa lyftkraften övergick samma värde antog funktionen ett högt värde. Effekten av detta var att lösningar där den negativa lyftkraften inte bibehölls straffades med ett högt värde på luftmotståndet. I den ursprungliga lösningen låg den negativa lyftkraften under straffvärdet och därför antog stegfunktionen värdet för luftmotståndet initialt.

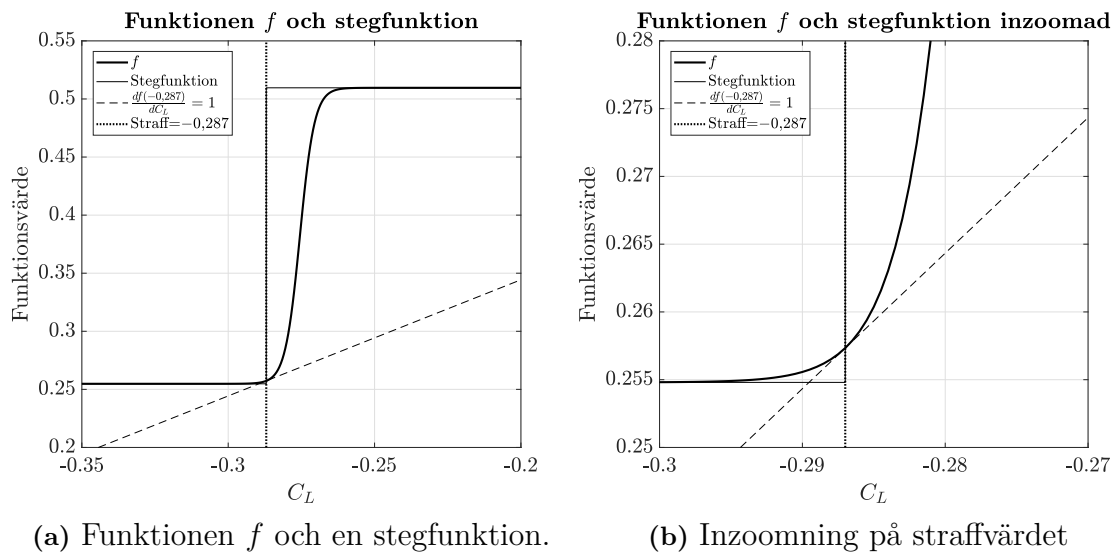
Eftersom en stegfunktion är diskontinuerlig och adjointmetoden är baserad på gradienter användes även en jämnare och kontinuerlig funktion,

$$f(C_L) = C_D \cdot \left(\frac{1}{1 + e^{-a \cdot (C_L + b)}} + 1 \right) \quad (3.1)$$

där a och b är konstanter och $a > 0$. Funktionen skapades genom att modifiera en sigmoid funktion så att funktionen antog värdet för C_D vid låga C_L och $2 \cdot C_D$ vid höga C_L . För att likna en stegfunktion krävdes det att funktionen snabbt växte mellan dessa nivåer. Därför ansattes $a = 400$. Utifrån grundsimuleringen ansattes straffvärdet till när $C_L = -0,287$, som är värdet på C_L för bil och vinge avrundat för denna simulering. Denna simulering hade också $C_D \approx 0,2548$. Om $C_L > -0,287$ skulle funktionen anta ett högt värde och om $C_L < -0,287$ skulle funktionen anta ett lågt värde. Eftersom funktionen är kontinuerlig behövdes en lämplig punkt väljas, på funktionen, där straffvärdet bedömdes vara. Därför ansattes att straffvärdet sker när

$$\frac{df(-0,287)}{dC_L} = 1, \quad \frac{d^2f(-0,287)}{dC_L^2} > 0. \quad (3.2)$$

Detta gav att $b = 0,2755$ för att förskjuta funktionen till rätt placering. Den resulterande funktionen presenteras i figur 3.15 tillsammans med en stegfunktion som har samma straffvärde.



Figur 3.15: Funktionen f i (3.1) och en stegfunktion mellan C_D och $2 \cdot C_D$, med diskontinuiteten placerad vid $C_L = -0,287$.

I figur 3.15 visar den tjocka svarta linjen funktionen (3.1) för det ursprungliga värdet på C_D , och den streckade linjen visar derivatan av f i $C_L = -0,287$. Den tunna svarta linjen visar en stegfunktion med samma straffvärde och nivåer. Notera att detta inte är den tidigare diskuterade stegfunktionen. Strafnivån för C_L visas med den prickade linjen.

Adjointlösaren användes sedan med en kostnadsfunktion som utgick från den *Expression Report* som skapats för ekvation (3.1).

3.2.5.2 Konvergens av adjointlösaren

Det finns inbyggda funktioner i STAR-CCM+ som hjälper för snabbare konvergens av adjointlösarens residualer. Dessa är extraval till adjointlösaren och heter *Restarted GMRES* och *Flexible GMRES* där *GMRES* står för *Generalized minimal residual method* [22]. *Flexible GMRES* användes i de fall där adjointlösaren inte konvergerade vid användning av *Restarted GMRES* och är alltså ett mer robust tillägg till adjointlösaren, men som däremot kan öka beräkningstiden [22]. För dessa två metoder måste antalet dimensioner av krylovrum och defektionskorrigeringsiterationer väljas. En ökning av dimensioner av krylovrum förbättrar konvergens och robustheten hos *GMRES*-metoderna medan defektionskorrigeringsiterationer används för att jämna ut felet [22].

I ett tidigare kandidatarbete [23] användes dessa metoder med olika inställningar på antalet dimensioner av krylovrum och defektionskorrigeringsiterationer vid simulering på en bilmodell. För denna modell kunde konvergens börja anas för 50 dimensioner av krylovrum och 3 defektionskorrigeringsiterationer. Konvergens skedde vid 100 dimensioner av krylovrum och 10 defektionskorrigeringsiterationer. Eftersom även det aktuella arbetet inkluderade simulering på en bil antogs det vara lämpligt att

börja med kombinationen 50 och 3, för att sedan testa 100 och 10. *Flexible GMRES* var också den metod som i huvudsak användes för att snabbt uppnå konvergens.

Adjointlösaren kunde ställas in på diskretisering enligt *1st-order* eller *2nd-order* [24]. Detta innebär att den använde angränsande celler respektive angränsande celler och den ytterligare ett steg bort. Grundinställningen är *2nd-order*. Vid användning av *1st-order* blev lösningen mindre noggrann, men mer robust. Den var lämplig att använda när *2nd-order* inte konvergerade. Båda dessa metoder användes.

Slutligen, när konvergens hade uppnåtts, beräknades ytkänsligheten för geometrin. Denna visualiserades sedan på vingen.

3.2.5.3 Kontrollpunkter och deformation av geometri

För att deformera geometrin behövdes ett antal kontrollpunkter placeras runt vingen. Dessa används för att beräkna förändringen hos geometrin.

Efter utplaceringen beräknades nätkänsligheten för att erhålla en vektor för varje kontrollpunkt. Denna indikerade hur geometrin skulle förändras för att maximera kostnadsfunktionen. Eftersom målet var att minimera ekvation (3.1) skedde deformationen i motsatta riktningen.

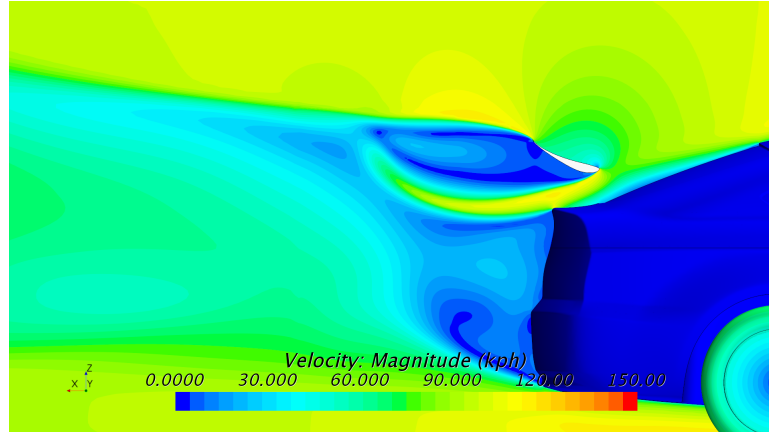
4

Resultat

I följande kapitel presenteras resultatet av implementeringen av de använda optimeringsmetoderna för optimering av den konstruerade bakvingen till DrivAer-modellen [3]. Först presenteras generella resultat, därefter resultatet från surrogatmodelleringen, följt av genetiska algoritmer och slutligen adjointmetoden.

4.1 Latin hypercube sampling

Det bästa resultatet från LHS, som användes för både surrogatmodellen och den genetiska algoritmen, visas i figur 4.1. Den aggressiva attackvinkeln, 41,19 grader, gör att flödesseparationen sker nästan direkt och skapar en stor vak med låg hastighet bakom vingen. Då flödesseparationen sker tidigt hinner inte luften accelerera längs ytan på vingen och därför blir hastigheten inte högre än ≈ 125 km/h.



Figur 4.1: Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L från LHS, för både surrogatmodellen och genetiska algoritmen, med $C_D = 0,347$ och $C_L = -0,424$.

4.2 Surrogatmodellering

Avsnittet som följer presenterar optimeringsresultat framtagna med hjälp av surrogatmodellen. Totalt har 40 simuleringar genomförts, varav 22 av dessa med höjd och attackvinkel som optimeringsvariabler. Resterande 18 simuleringar inkluderade även, utöver attackvinkel och höjd, longitudinellt led som optimeringsvariabel. Alla simuleringar har uppnått konvergens, dock med lite olika marginal. Vissa simuleringar hade konvergerat med lägre precision än andra. Resultatet för vingens position anses också ha konvergerat mot ett specifikt höjdintervall, vinkelintervall och position i longitudinellt led.

Efter att de första 16 simuleringarna genomförts observerades det att medianen konvergerat mot en höjd på 0,2 meter. I tabell 4.1 redovisas höjder från surrogatmodellens tre sista medianer av de 16 första simuleringarna. Då 0,2 meter är projektets minimihöjd, enligt tabell 4.2a, valdes denna att korrigeras till att även täcka in lägre höjder. Tabell 4.2b presenterar det korrigerade intervallet för höjd och vinkel.

Tabell 4.1: Konvergensintervall för surrogatmodellering.

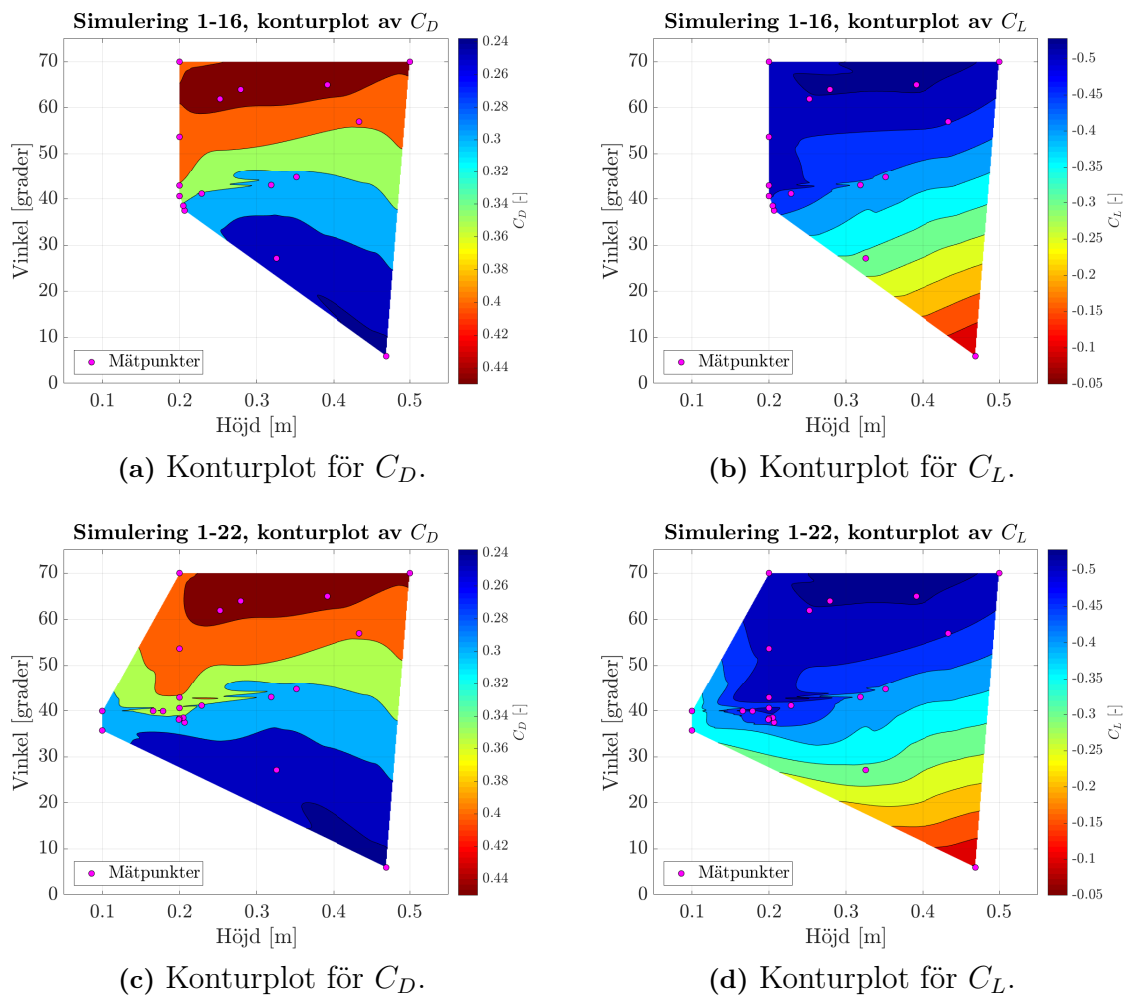
Median nummer	Höjd [m]
4	0,2000
5	0,2066
6	0,2049

Med en minimihöjd på 0,1 meter krävs det att vingens maximala attackvinkel minskas. Detta för att vid en aggressiv attackvinkel i kombination med en låg höjd riskerar vingen att skära igenom bilens baklucka.

Tabell 4.2: Gränser för intervallet.

(a) Höjd- och vinkelintervall från start.			(b) Korrigerat höjd- och vinkelintervall.		
	Höjd [m]	Vinkel [°]		Höjd [m]	Vinkel [°]
Min	0,20	5	Min	0,10	5
Max	0,50	75	Max	0,25	40

I figur 4.2 redovisas konturplottar. Konturplottarna skapas genom egenskriven kod i MatLab. Koden använder kubisk interpolering med hjälp av funktionen *griddata*. Konturplottarna ger en bra överblick hur optimeringsvariablerna, höjd och attackvinkel, förhåller sig till C_D samt C_L . Figur 4.2 innehåller fyra stycken plottar varav figur 4.2a och 4.2b visar hur höjden och attackvinkeln förhåller sig till C_D respektive C_L för de 16 första simuleringarna. I dessa figurer observeras att en stor del av mätpunkterna hamnar på minimihöjden från tabell 4.2a.



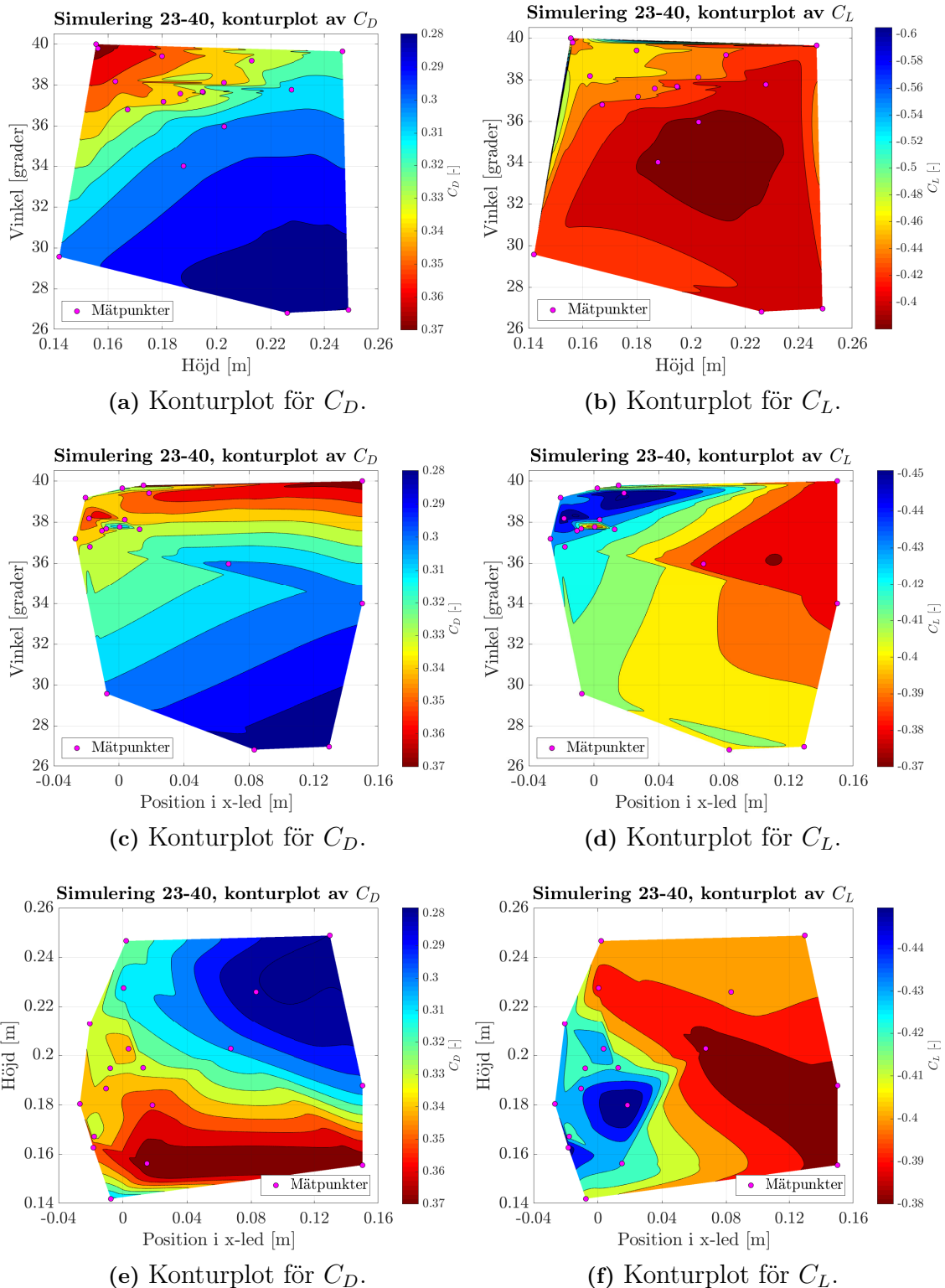
Figur 4.2: Konturplottar för optimeringsvariabler i korrelation med C_D och C_L .

Intervallen för figur 4.2a och 4.2b är enligt tabell 4.2a. Mätpunkterna i figur 4.2 visar de simulerade värdena på optimeringsvariablerna, givna av surrogatmodellen, samt inom vilka intervall värdena för C_D , figur 4.2a och 4.2c, och C_L , figur 4.2b och 4.2d, antar. Vidare utfördes ytterligare 6 simuleringar där det nya intervallet, tabell 4.2b, är inkluderat. Resultaten för samtliga 22 simuleringar observeras i figur 4.2c och 4.2d. Dessa styrker det tidigare antagandet om att den optimala höjden för vingen var lägre än 0,2 meter, då mätpunkterna rör sig mot en lägre höjd.

I figur 4.2c och 4.2d observeras konvergens för optimeringsvariablerna. Att variablerna konvergerat mot ett visst värde observeras genom att större delen av mätpunkterna grupperas kring en höjd på 0,19 meter och en vinkel på 38 grader. Vidare undersöktes kraven på konvergens enligt rubrik 3.2.2 för dessa.

Studien valdes här att utökas till ytterligare en optimeringsvariabel, vingens position i longitudinellt led, sett från sidan av bilen, se rubrik 3.2, figur 3.7. Fortsatt i rapporten benämns variabeln longitudinellt led som x-led. Det utfördes totalt 18 simuleringar med x-led som optimeringsvariabel tillsammans med de tidigare variablerna. Resultaten observeras i figur 4.3.

4. Resultat



Figur 4.3: Konturplottar för optimeringsvariabler i korrelation med C_D och C_L .

Efter att alla punkter studerats och konturplottarna kontrollerats är det tre simuleringar som anses ge de bästa värdena på C_D efter kravet på C_L . Tabell 4.3 redovisar värden på optimeringsvariablerna för de tio sista simuleringarna som utfördes

i STAR-CCM+ med hjälp av surrogatmodellen. Nedan presenteras också de tre resultaten som anses uppfylla kravet på bibehålldet C_L med så lågt C_D som möjligt. Dessa är markerade med ljusgrön färg i tabellen. Samtliga simuleringar med tillhörande resultat presenteras i appendix B, figur B.1.

Tabell 4.3: Resultat från surrogatmodellens tio sista simuleringar. De bästa simuleringarna är markerade med ljusgrön färg.

Nr	Höjd [m]	Vinkel [°]	X-led [m]	Typ	C_D -värde	C_L -värde
31	0,1950	37,65	0,0127	MEDIAN	0,336	-0,418
32	0,1627	38,18	-0,0183	STD	0,358	-0,444
33	0,1948	37,68	-0,0078	MEDIAN	0,338	-0,416
34	0,2489	26,97	0,1296	STD	0,280	-0,399
35	0,2028	35,97	0,0673	RAND	0,309	-0,378
36	0,1799	39,41	0,0185	STD	0,358	-0,449
37	0,1866	37,58	-0,0104	MEDIAN	0,344	-0,428
38	0,1804	37,19	-0,0267	MEDIAN	0,344	-0,425
39	0,2129	39,19	-0,0205	MEDIAN	0,338	-0,420
40	0,2027	38,12	0,0036	MEDIAN	0,345	-0,428

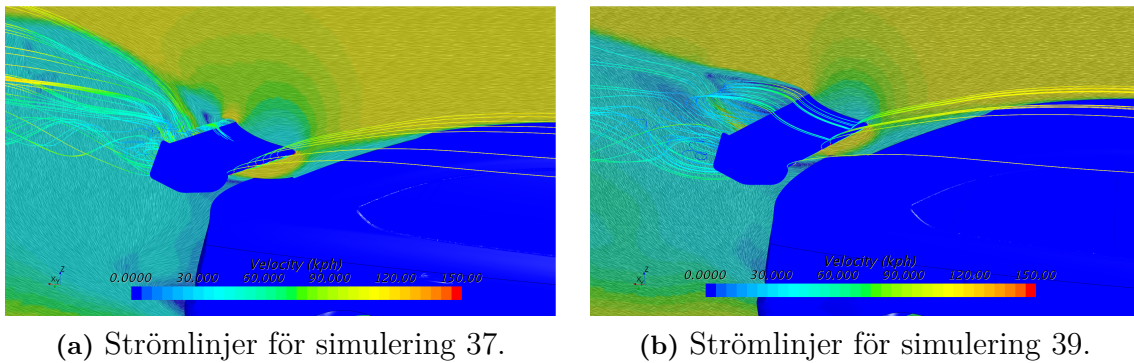
Det kan observeras att för samtliga medianer i tabell 4.3 ligger C_L -värdet inom ett intervall av $-0,416$ och $-0,428$ vilket anses uppfylla kravet på $C_L \approx -0,425$. Detta är för att CFD inte har perfekt precision och därför blir alla värden approximativa. Vidare befinner sig värdena för C_D inom ett intervall av $0,336$ och $0,344$.

Anledningen till att dessa tre simuleringar anses ge dem bästa resultaten är på grund av C_L -värdet varierar med en precision på tusendelsnivå runt det önskade värdet $-0,425$. Samtidigt befinner sig samtliga tre värden för C_D under $0,345$ vilket inte erhållits tillsammans med så starka C_L -värden tidigare.

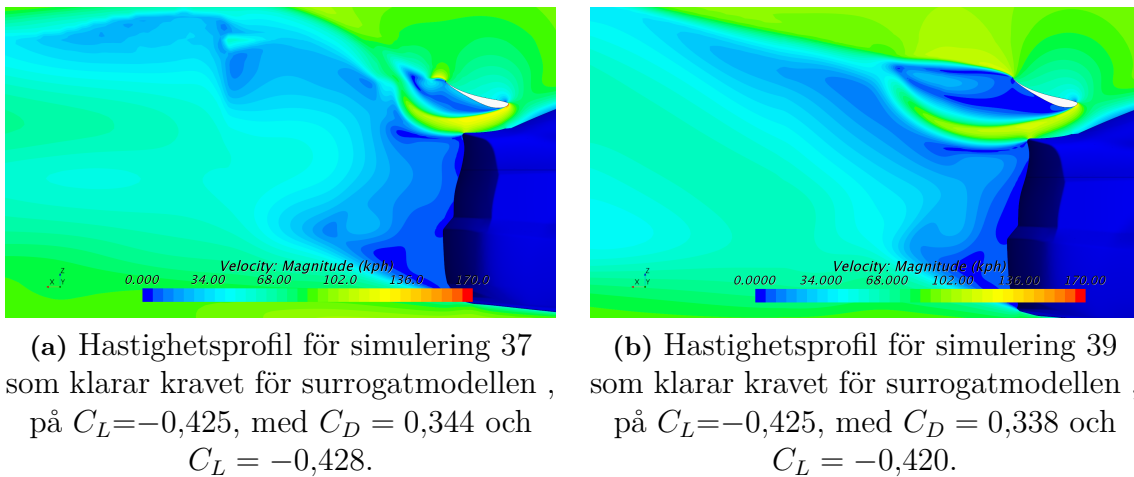
Residualer för simulering nummer 39 presenteras i appendix B, figur B.1. Residualerna för de andra två simuleringarna skiljer sig inte nämnvärt mot denna och presenteras därför inte. Tillhörande plott för y^+ på bilen presenteras också i appendix B, figur B.2, och skiljer sig inte heller mellan de olika simuleringarna.

4.2.1 Strömlinjer och vak

I figur 4.4 presenteras strömlinjer i vaken och kring vingen för två av de bästa simuleringarna. I figur 4.5 observeras hastighetsprofilen för luftströmmen. Dessa figurer ger en överblick i hur luftströmmarna beter sig bakom bilen och kring vingen. Hastigheterna för luftströmmarna observeras också i samma figurer. Den sista simuleringen, nummer 40, skiljer sig inte speciellt från nummer 39 och presenteras därför inte med bild.



Figur 4.4: Strömlinjer för simulering 37 och 39.

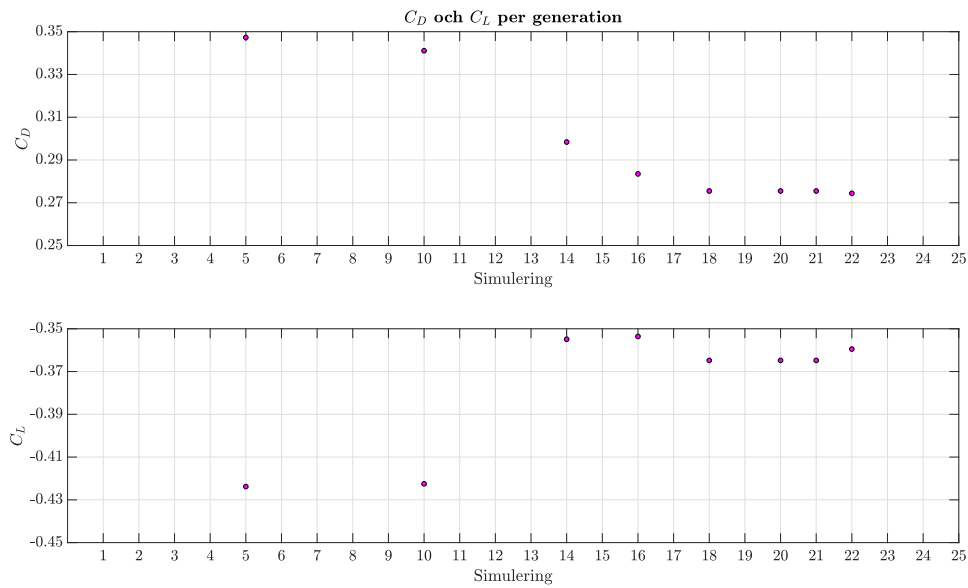


Figur 4.5: Hastighetsprofil för simulering 37 och 39.

4.3 Genetiska algoritmer

Totalt kördes 22 simuleringar varav 21 simuleringar konvergerade. För den simulering som inte konvergerade ändrades höjden för vingen en tiondels millimeter vilket ledde till konvergens. Totalt genomfördes åtta generationer med fem individer vardera. De individer som inte simulerades är alltså dubletter av tidigare simulerade individer. I figur 4.6 visas resultat för den bästa individen i varje generation.

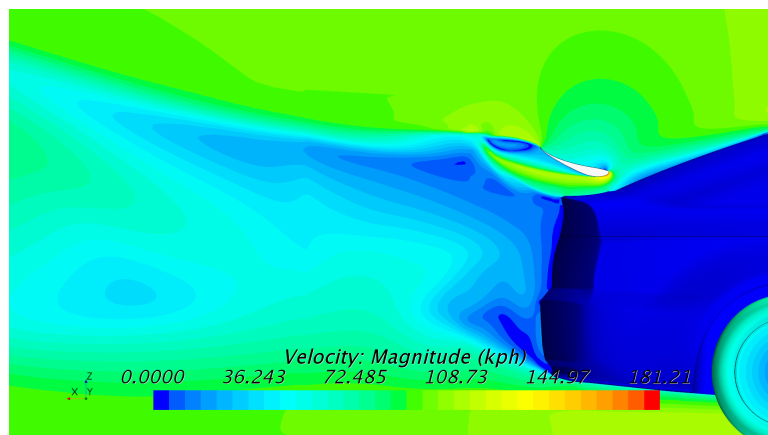
4. Resultat



Figur 4.6: C_D och C_L för den bästa individen varje generation.

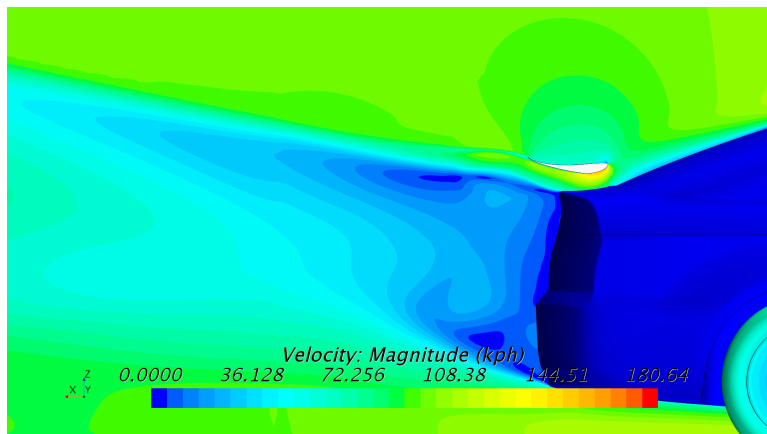
Med hjälp av LHS stabiliserar sig den genetiska algoritmen fort mot låga C_D -värden. Efter 14 simuleringar hittades variabler för $C_D < 0,3$. Därefter går det betydligt långsammare, men C_D -värdet minskar ändå successivt. I det här arbetet var det totala antalet individer 40, vilket är lågt relativt det rekommenderade minimumantalet 3000 [16]. Därav går det inte att säkerställa att lösningen har hittat ett globalt minimum. Istället förväntas det finnas konfigurationer som är bättre.

I generation två hade den bästa lösningen ett C_D -värde på 0,3411 och ett C_L -värde på $-0,4225$ och visas i figur 4.7. Hastighetsprofilen är lik den i figur 4.1, dock med en mindre vak och något senare flödesseparation. Detta resulterar i ett ungefärligt bibehållet C_L och betydligt lägre C_D .



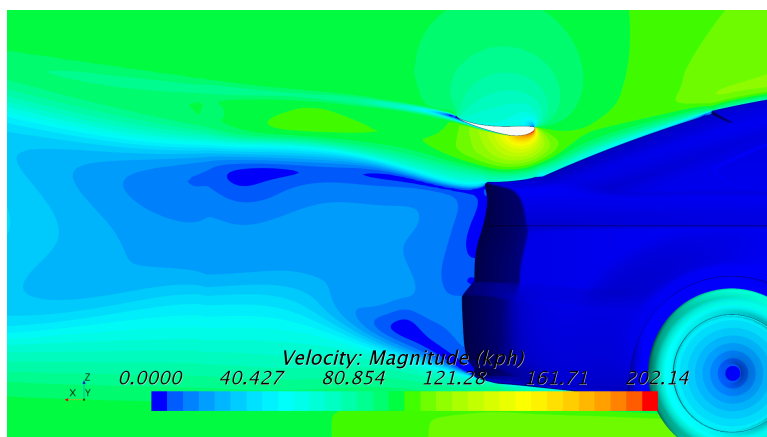
Figur 4.7: Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation 2.

I generation fyra hittar algoritmen en profil med lägre höjd och lägre vinkel, se figur 4.8. Här sker flödesseparationen längre bak på vingen, samt högsta hastigheten är ≈ 25 km/h snabbare. C_D -värdet har minskat betydligt, men till priset av att C_L har ökat.



Figur 4.8: Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation 4.

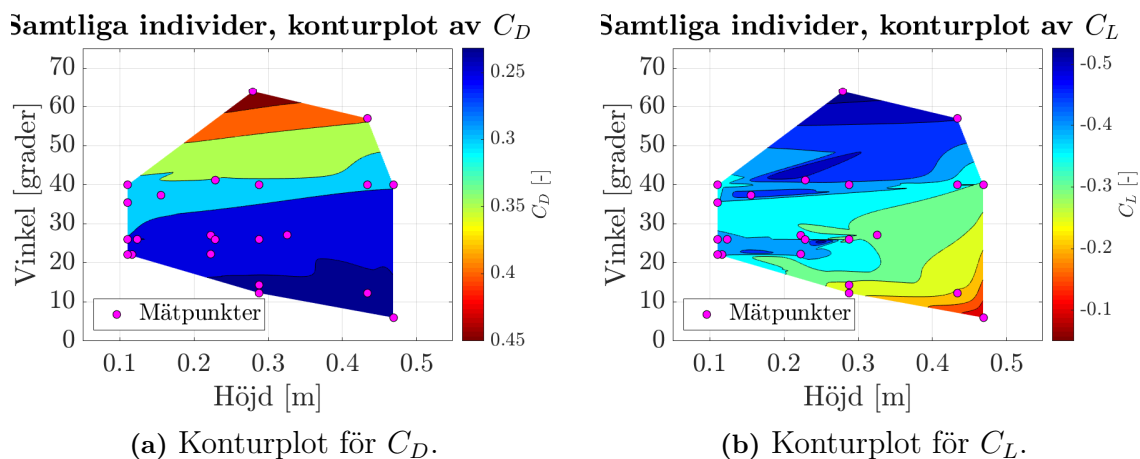
I den sista generationen som kördes, generation åtta, hittade algoritmen den bästa lösningen algoritmen genererat, se figur 4.9. Detta tyder på att algoritmen hade funnit bättre individer om den hade fått fortsätta.



Figur 4.9: Hastighetsprofil för den vinge med lägst C_D och som klarar kravet på C_L i generation åtta.

Efter bara några generationer kräver den genetiska algoritmen färre simuleringar per generation då många individer blir identiska gentemot tidigare individer. Detta innebär att totala antalet individer kan öka fort utan att många simuleringar krävs. Jämfört med referensbilen som inte har någon vinge lyckades den genetiska algoritmen minska C_L med 0,3644 samtidigt som C_D -värdet ökade med 0,0445 eller 19,4 %.

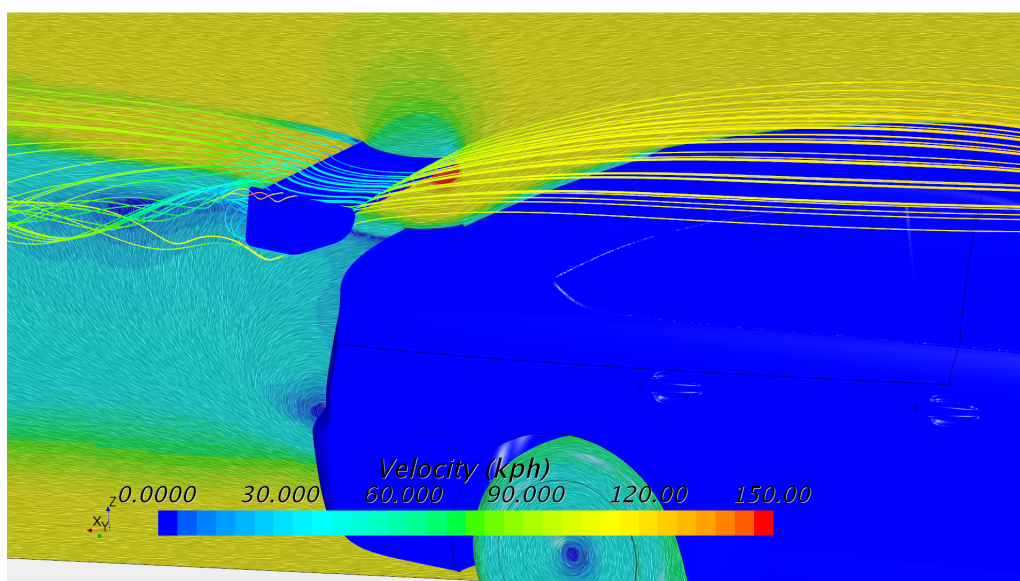
I figur 4.10 redovisas konturplottar likt de konturplottar i figur 4.2. Dessa konturplottar visar hur C_D och C_L förhåller sig till höjd och vinkel, samt vilka mätpunkter den genetiska algoritmen undersökt.



Figur 4.10: Konturplottar för optimeringsvariabler i korrelation med C_D och C_L .

4.3.1 Strömlinjer

Strömlinjerna, se figur 4.11, visar att flödet kring den bästa genererade vingen är jämnt med liten återcirkulering, vilket indikerar lågt luftmotstånd och därmed låg luftmotståndskoefficient.



Figur 4.11: Visualisering av strömningslinjer kring vingen med lägst C_D som klarar kravet på C_L .

4.4 Adjointmetoden

I följande avsnitt presenteras resultaten från adjointmetoden. Först presenteras resultatet från användning av olika kostnadsfunktioner. Därefter presenteras resultatet från olika inställningar för att uppnå konvergens för adjointlösaren. Slutligen presenteras ytkänsligheten för vingen.

4.4.1 Kostnadsfunktion

Flertalet tester har utförts på fristående vingar eftersom svårigheter med att få adjointlösaren att konvergera har funnits. Den första funktionen som användes i en *Expression Report* var en if-sats som hade anpassats för att kunna implementeras i STAR-CCM+. Användandet av en sådan stegfunktion fungerade när den negativa lyftkraften befann sig under diskontinuiteten, alltså att funktionen antog värdet för luftmotståndet. Om lyftkraften gick över gränsen och funktionen straffades genom att luftmotståndet fick ett högt värde, så kunde inte adjointlösaren konvergera. Vid användning av en jämnare och mer kontinuerlig funktion som en *Expression Report*, ekvation (3.1), tycks adjointlösaren kunna konvergera för ett av de testade fallen med vingen i 3.1.2 och på DrivAer-modellen. Konvergens diskuteras vidare i kapitel 4.4.2.

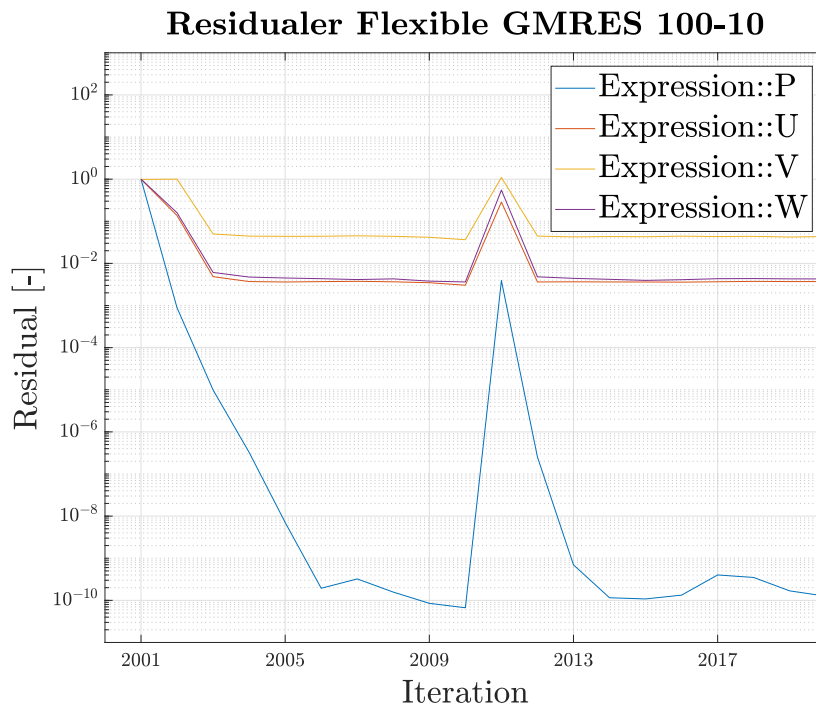
4.4.2 Konvergens

I figurer i detta avsnitt benämns antalet dimensioner av krylovrum och defektionskorrigeringsiterationer på formen 100-10.

Vid användning av 50 dimensioner av krylovrum och 3 defektionskorrigeringsiterationer tycks konvergens ha uppnåtts kring värdet 1. Detta för både *Restarted GMRES* och *Flexible GMRES*. Residualplottar för dessa presenteras i appendix D, figur D.1 respektive D.2. *Flexible GMRES* tycks konvergera snabbare än *Restarted GMRES*. I båda dessa fall användes ett något annat värde på b i ekvation (3.1) än beskrivet i kapitel 3.2.5.1, här användes $b = 0,27543$. Antalet core-timmar som användes var ungefär 400 respektive 450.

I appendix D, figur D.3, presenteras residualerna för fallet med användning av 100 dimensioner av krylovrum och 10 defektionskorrigeringsiterationer. Adjointlösaren var inställd på *Flexible GMRES* och med diskretisering på *2nd-order*. Residualerna tycks konvergera kring värdet 1. Detta är ett liknande beteende som i figur D.1 och D.2. Antalet core-timmar som krävdes var cirka 1300.

Fallet med 100 dimensioner av krylovrum och 10 defektionskorrigeringsiterationer och adjointlösaren inställd på diskretisering enligt *1st-order* presenteras i figur 4.12. Residualerna tycks börja konvergera under de första 10 iterationerna. Därefter genomfördes 10 ytterligare iterationer vilket illustreras av att residualerna steg upp till en hög nivå vid iteration 2011 för att sedan återigen börja konvergera. Troligtvis skulle fler iterationer behövas för att säkerställa konvergens. Intressant är att residualerna tycks konvergera mot väldigt olika värden, tre runt 10^{-2} och en på 10^{-10} . Totalt krävdes det ungefär 1600 core-timmar för 20 iterationer.

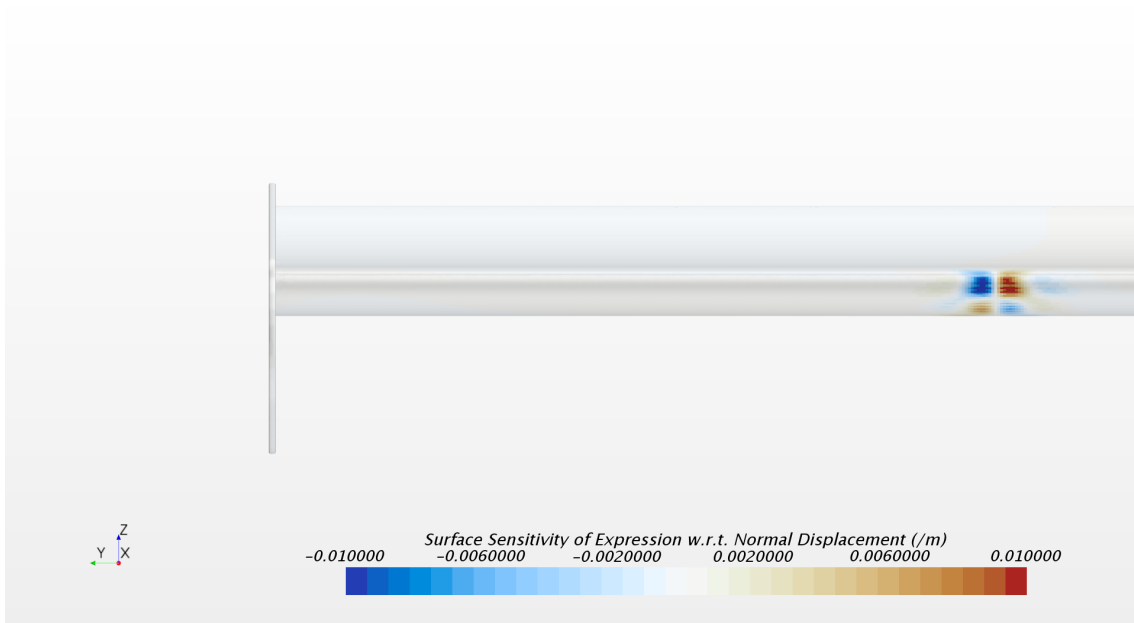


Figur 4.12: Residualer för 100-10 och *1st-order*. Notera att vid iteration 2011 ökar residualerna snabbt för att sedan minska igen vilket skedde när adjointlösaren fortsatte.

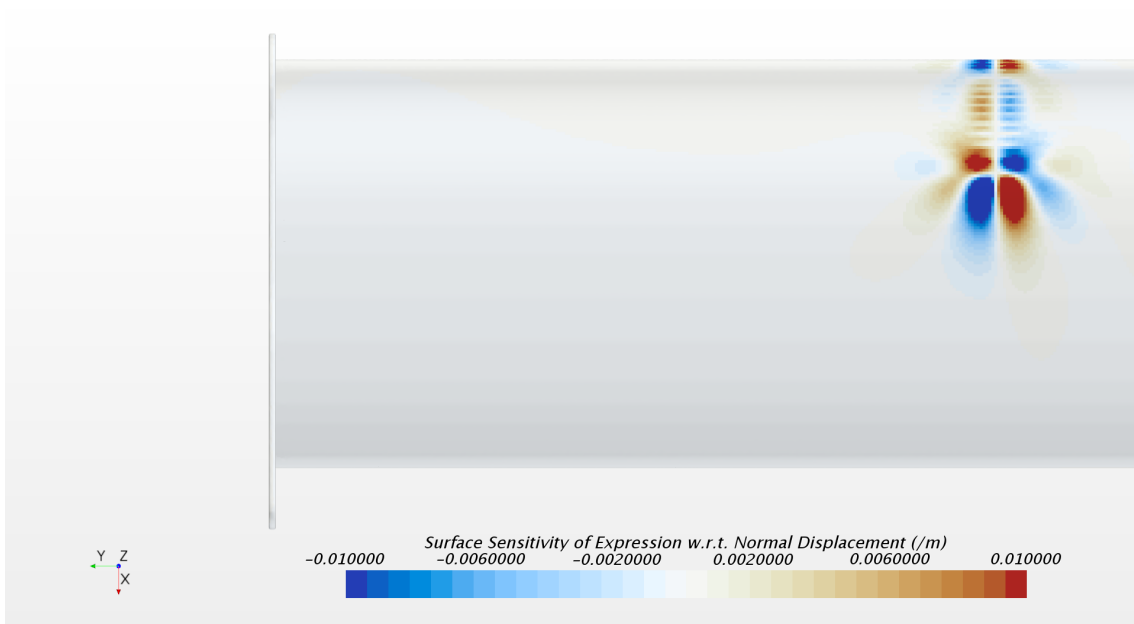
4.4.3 Ytkänslighetsdata

I STAR-CCM+ kan ytkänsligheten beräknas för att avgöra vilka områden som behöver förändras. Eftersom endast vingen är av intresse beräknades bara ytkänsligheten för vingen. Resultatet presenteras i figur 4.13 och med en *Sensitivity Smoothing Factor* på 100. För att minska värdet på ekvation (3.1) ska röda områden förskjutas utåt och blåa områden inåt längs med ytans normal. Notera att endast en liten del av vingen har fått värden på ytkänsligheten. Troligtvis är detta inte ett korrekt resultat eftersom hela vingen borde kunna förändras, inte bara en liten del.

4. Resultat



(a) Vingen framifrån.



(b) Vingen underifrån.

Figur 4.13: Ytkänsligheten för vingen med avseende på ekvation (3.1). Notera att bara en liten del av vingen ska förändras enligt beräkningarna.

5

Diskussion

I kapitlet som följer diskuteras de resultat som kunnat dras av de tre optimeringsmetoderna samt för- och nackdelar med dessa.

5.1 Surrogatmodellen

I följande avsnitt diskuteras vad som låg till grund för beslut som togs under arbetets gång med surrogatmodellen samt för- och nackdelar med optimeringsmetoden. Vidare presenteras rekommendationer för fortsatt arbete och en utvärdering av modellen som helhet.

I arbetet med att optimera bakvingen genom surrogatmodellering konstaterades det tidigt i arbetsprocessen att intervallen för höjden och attackvinkeln var tvungna att korrigeras. Surrogatmodellen hittade snabbt en optimal höjd på 0,2 meter över bakluckan på bilen. Då detta också var minimihöjden för det första valda intervallet föreslog inte surrogatmodellen något värde under denna höjden. Detta innebar att den optimala höjden för att hitta ett så litet C_D som möjligt, med ett bibehållet $C_L \approx -0,425$, kunde ligga under intervallet. När modellen givit tre medianer på minimihöjden beslutades det att korrigera intervallet för att utöka lösningsrymden ytterligare.

Efter korrigering av intervallen konvergerade höjden och attackvinkeln snabbt, se rubrik 4.2. Vidare beslutades det att utöka antalet optimeringsvariabler från två till tre genom att även optimera vingens position i x-led sett till bilens baklucka, se figur 3.7. Detta medförde att surrogatmodellen föreslog medianer som inte var i närheten av de tidigare sett till höjd och attackvinkel. Då modellen lärde sig av redan existerande data konvergerade höjden och attackvinkel igen medan optimeringsvariablen, x-led, fortfarande varierade. För att hitta den mest optimala kombinationen av alla optimeringsvariablerna hade fler simuleringar krävts.

Vidare har det konstaterats att kravet på C_L ansattes alldeles för strängt. I figur 4.5 observeras en stor vak bakom vingen, för respektive simulering. Attackvinkeln blir, på grund av kravet för C_L , för stor och separation uppkommer därför bakom vingen. Separationen bidrar till ett högt värde på C_D , vilket är ineffektivt. Hade kravet för C_L minskats med 0,0265 hade värden på C_D kunnat minskas från 0,3378 ner till 0,2798, vilket är en minskning av C_D på 0,058. Förbättringen av C_D blir därmed mer än dubbelt så stor som försämringen av C_L vilket kan övervägas. Simulering nummer 34 antar dessa värdena på motståndskoefficienterna och vaken för denna

observeras i appendix B, figur B.3. Det kan här observeras att vaken är mer eller mindre obefintlig och inget nämnvärt separationsflöde uppkommer. Med kravet för C_L sänkt hade surrogatmodellen kunnat hitta positioner liknande och bättre än detta exempel, vilket hade genererat ett bättre resultat än de tre aktuella i tabell 4.3.

5.1.1 Utvärdering och rekommendationer

Surrogatmodellen har stor potential för vidareutveckling. Vid ett eventuellt vidarearbete med detta projekt rekommenderas optimering med fler variabler från början. Eftersom optimeringsvariabeln x -led fortfarande varierar kring punkten noll hade det varit optimalt att ha med den tidigare och därmed låta surrogatmodellen lära sig av den också. Att starta en ny simulering tog cirka en timma vilket är anledningen till att det rekommenderas att automatisera det manuella arbetet. En automatisering av denna process hade sparat mycket tid, speciellt då fler än tre optimeringsvariabler är aktuellt.

5.2 Genetiska algoritmer

I detta avsnitt diskuteras resultaten från den genetiska algoritmen som implementerades under projektet. För- och nackdelar kommer tas upp, samt resonemang om tagna beslut kommer att diskuteras. Slutligen presenteras möjliga vidareutvecklingar av den genetiska algoritmen.

Av samma anledningar som är diskuterade i avsnitt 5.1 valdes det att även inom den genetiska algoritmen ändra intervallet för höjd och attackvinkel. Denna korregering antas också bidra till att den genetiska algoritmen inte kunde hitta individer med det första tänkta C_L -kravet på $-0,425$. Anledningen är att attackvinklar över 40 grader inte längre fick genereras, vilket resulterade i att antalet konfigurationer som klarade C_L -kravet minskades betydligt. För att snabbare få fram resultat användes $C_L = -0,350$ som gräns. Hade fler individer eller generationer kunnat användas hade den högre gränsen inte varit något problem, då algoritmen förr eller senare hade hittat en lämplig konfiguration. En möjlig lösning för att klara den hårda gränsen snabbare är att modifiera algoritmen så att värden närmre gränsen straffas mindre, och på så vis stega sig fram till ett C_L -värde som uppfyller kravet. I detta fall skulle användningen av en modifierad sigmoid funktion liknande den för adjointmetoden, se kapitel 3.2.5.1, vara fördelaktigt. Det skulle innebära att straffet för nivån på C_L får en jämnare övergång och att strafffunktionen blir mer kontinuerlig, det vill säga att värden närmre gränsen straffas mindre.

5.2.1 Mutering

På grund av det låga totala antalet individer valdes mutationschansen till 20 % vilket är betydligt högre än rekommenderade 2-5 %. På grund av den begränsade mängden generationer antogs det att lösningsrymden inte skulle kunna täckas bra med en låg mutationschans. Dessutom fanns det endast 10 variabler per generation, vilket innebär att i genomsnitt muterades två variabler i varje generation. Med fler generationer eller ett större antal individer per generation hade en lägre mutationschans kunnat användas för att minimera risken att fördelaktig avkomma muteras bort.

5.2.2 Utvärdering och rekommendationer

En nackdel med genetiska algoritmer inom fordonsaerodynamik är att utvärderingen av varje individ är beräkningsmässigt tung, vilket försvårar möjligheterna att utvärdera tillräckligt många individer för att få ett säkert resultat. Eftersom mycket baseras på slump behövs det många individer för att säkerställa att ett optimalt resultat har uppnåtts.

Den genetiska algoritmen är mycket enkel att implementera. Förarbetet går fort för varje unikt användningsområde. Det går även enkelt att använda flera variabler. Eftersom endast två variabler testades i studien och vingprofilen hölls konstant är det potentiellt enklare att ta fram en modell och utföra fysiska experiment och på så vis få fram en graf för hela lösningsrymden. När fler variabler undersöks blir den genetiska algoritmen möjligen enklare att implementera än en fysisk modell, speciellt om geometriförändringar används som variabel.

Metoden lämpar sig väl för fullständig automatisering. För att metoden ska fungera bra är detta ett krav snarare än ett önskemål då uppstart av nya simuleringar är tidskrävande, precis som för surrogatmodellen.

Antalet simuleringar som kördes per generation är betydligt lägre än vad som först väntades då endast nya värden behöver simuleras. Redan vid generation fyra behöver endast två av fem individer simuleras eftersom övriga individer är dubletter från tidigare generationer. Två förbättringsidéer kan dras från detta. Den första är att antalet individer per generation hade kunnat ökas för att maximera produkten av individer och generationer. Det går då att statistiskt ta fram hur många dubletter som bör fås. Den andra förbättringen är att skriva ett skript för att upptäcka dubletter så att de inte simuleras, detta krävs vid fullständig automatisering.

5.3 Jämförelse mellan surrogatmodellering och genetiska algoritmer

En jämförelse mellan den genetiska algoritmen och surrogatmodellen har utförts. Tanken med denna jämförelse är att ställa optimeringsmetoderna mot varandra och jämföra vilka optimala värden på luftmotståndskoefficienten, C_D , som genereras med givna variabler. Jämförelsen skall också ta hänsyn till vilka värden på C_L som erhålls för respektive optimeringsmetods bästa resultat och hur luftflödet ser ut över vingen för respektive resultat.

Vidare kommer också tiden för att få metoden till att konvergera mot ett resultat jämföras och slutsatser kring vilken utav dessa metoder som lämpar sig bäst för optimering av en bakvinge, enligt fakta framtagna i detta kandidatarbete, kommer dras.

5.3.1 Strömlinjer och vak

Som observeras i figur 4.5 och figur 4.9 är vaken för vingen betydligt mindre då den genetiska algoritmen används som optimeringsmetod. Detta är för att kravet på C_L skiljer sig mellan metoderna. Enligt rubrik 4.3 används $C_L = -0,350$ på grund av att den genetiska algoritmen hade svårt att generera värden som uppfyller kravet $C_L = -0,425$. Med ett mildare krav på C_L kan optimeringsmetoden generera mindre aggressiva attackvinklar, vilket medför mindre separation och därmed lägre C_D .

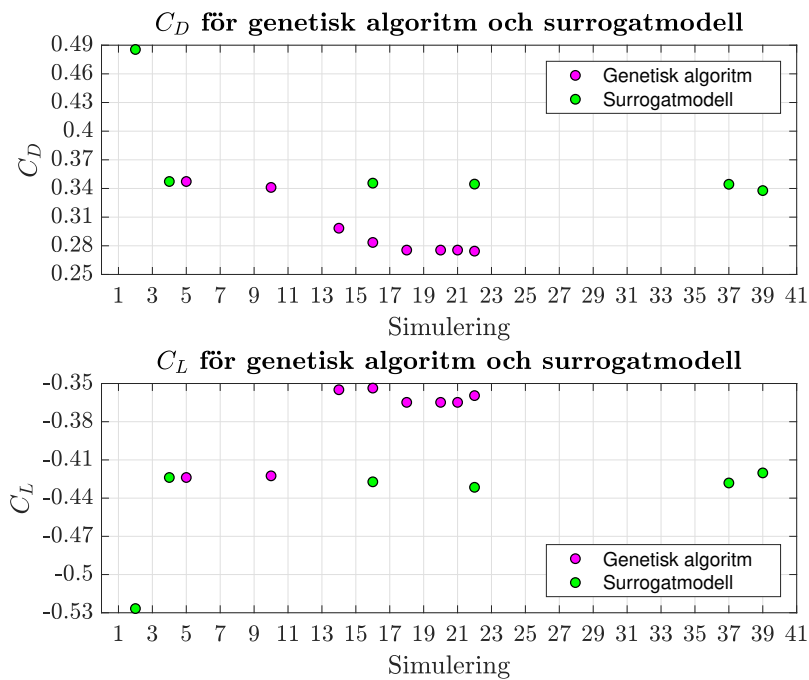
5.3.2 Luftmotståndskoefficienten, C_D

Tabell 5.1 jämför surrogatmodellens bästa värden för C_D mot den genetiska algoritmens. Givet kraven på C_L för respektive optimeringsmetod redovisas resultatet nedan.

Tabell 5.1: Jämförelse av luftmotståndskoefficienten hos surrogatmodellen och den genetiska algoritmen.

Surrogatmodellen		Genetiska algoritmen	
C_D	C_L	C_D	C_L
0,344	-0,428	0,274	-0,360
0,338	-0,420	0,276	-0,365
0,345	-0,428	-	-

Samma slutsatser som antogs under rubrik 5.3.1 antas även här. Anledningen till att värdet på C_D är lägre för den genetiska algoritmen är för att kravet på C_L är mildare. Därmed genererar denna optimeringsmetod lägre värden på C_D än vad surrogatmodellen gör. Det observeras i figur 5.1 att surrogatmodellen hittar optimala värden snabbare än den genetiska algoritmen.



Figur 5.1: Jämförelse för konvergens av C_D , givet kravet på C_L , för respektive metods dittills bästa simulering.

5.3.3 Summering

Den genetiska algoritmen är mindre beräkningsmässigt krävande mellan varje simulering. Surrogatskripten tar cirka 30 sekunder att generera nya värden på optimeringsvariablerna medan den genetiska algoritmen tar 0,07 sekunder. Dock är denna skillnaden försumbar i detta projekt då simuleringarna i STAR-CCM+ tar cirka 250 core-timmar.

Surrogatmodellen har konvergerat mot ett litet höjd- och vinkelintervall. Det har inte den genetiska algoritmen gjort. Med denna informationen, tillsammans med tidigare redovisade bilder och tabeller, kan slutsatsen att surrogatmodellen är den optimeringsmetod som lämpar sig bäst för just detta projekt dras.

5.4 Adjointmetoden

I avsnittet som följer diskuteras för- och nackdelar med adjointmetoden. Vidare utvärderas metoden i sin helhet som optimeringsmetod och rekommendationer för fortsatt arbete presenteras.

Adjointmetoden optimerade till skillnad från surrogatmodellen och den genetiska algoritmen inte höjd och vinkel hos vingen. Anledningen var att det skulle krävas externa skript för att kunna genomföras. Istället valdes det att optimera vingens geometri. Målet med optimeringen var samma som för de andra två metoderna, att

bibehålla C_L och minska C_D . För att göra detta kunde inte de inbyggda funktionerna i STAR-CCM+ användas. Normalt minskar C_L om minskning av C_D genomförs. Därför skapades egna kostnadsfunktioner som skulle undvika detta genom att straffa lösningar som gav ett för högt värde på C_L . Straffet innebar att funktionen antog ett högt värde för C_D . En av de testade metoderna var att använda en if-sats i en *Expression Report* som skapade en stegfunktion med detta beteende. Problemet som uppstod med denna stegfunktion var att om lyftkraften gick över diskontinuiteten så divergerade adjointlösaren. Troligtvis berodde detta på att stegfunktionen inte är kontinuerlig vilket därför kan orsaka problem, eftersom adjointmetoden är gradientbaserad. För att försöka undvika detta problem användes en modifierad sigmoid funktion, ekvation (3.1), för att efterlikna stegfunktionen, men kontinuerlig och jämnare. Denna kunde adjointlösaren lösa och konvergens tycks kunna uppnås för vingen på DrivAer-modellen. Detta är också rimligt eftersom en kontinuerlig och jämn funktion är möjlig att derivera vilket gör det lättare för adjointlösaren att tillämpa den.

Konvergens är svårt att uppnå med adjointlösaren. I STAR-CCM+ finns olika metoder som kan hjälpa till med detta. Två av dessa är *Restarted GMRES* och *Flexible GMRES* som kan accelerera konvergensen, men de är också beräkningstunga. I figur 4.12 presenteras det fall som hade bäst konvergens av fyra testade. Den använde ungefär 1600 core-timmar för 20 iterationer och har inte konvergerat. Den tycks ha påbörjat att konvergera, men för att säkerställa konvergens skulle adjointlösaren ha behövt köras längre. Från de 10 första iterationerna kördes ytterligare 10 iterationer men residualerna ökade mycket innan de återigen minskade och tycktes konvergera. Möjligtvis hade en bättre konvergens kunnat uppnås om alla 20 iterationer genomförts i följd. Ett möjligt problem är att flödeslösningen inte konvergerat tillräckligt. Detta kan påverka hur bra konvergens adjointlösaren kan uppnå.

Det krävdes dessutom att adjointlösarens diskretisering ställdes in på *1st-order* för att få någon typ av konvergens. *1st-order* ger en mer robust lösning men är mindre noggrann. Detta skulle kunna vara ett problem men för det aktuella fallet med bakvingen är det troligtvis viktigare att kunna uppnå konvergens än att lösningen har en hög noggrannhet. Beräkning av ytkänsligheten är beroende av en väl konvergerad flödes- och adjointlösning för att fungera. Ytkänsligheten behövs för att kunna deformera geometrin, för att sedan köra flödeslösaren på den nya geometrin. Därför antas det vara mer lämpligt att använda den mindre exakta diskretiseringen om det är endast den som kan ge konvergens.

Ytkänsligheten beräknades utifrån adjointlösningen i figur 4.12 och resultatet presenteras i figur 4.13. Denna lösning är rimligtvis ofysikalisk då det endast är en liten del på undersidan av vingen som ska förändras. Väntat resultat var att förändringen skulle ske över hela vingen och inte som den aktuella beräkningen visar. Troliga förklaringar till detta är att inte tillräcklig konvergens uppnåtts i flödes- och adjointlösaren. Adjointlösningen i figur 4.12 hade som diskuterat tidigare troligtvis inte konvergerat tillräckligt bra.

5.4.1 Utvärdering och rekommendationer

Adjointmetoden har visat sig vara svårhanterlig, speciellt när det gäller adjointlösaren och dess konvergens. En orsak till detta är att kostnadsfunktionen inte får bestå av en diskontinuitet och att för att uppnå konvergens behövs stor beräkningskraft.

Adjointmetoden har, för denna typ av implementering, potential för vidareutveckling i framtida forskning. För att få en bättre förståelse av hur adjointmetoden på bästa sätt ska implementeras i ett fall som detta behöver konvergensen av adjointlösaren noggrannare analyseras. Kontrollpunkters placering och antal bör utvärderas mer noggrant inför framtida studier om detta steg är av intresse för att ytterligare se hur adjointlösaren påverkas.

En av adjointmetodens starka sidor är att den täcker en del av optimeringsrymden som inte surrogatmodellen eller den genetiska algoritmen täcker. Att adjointmetoden i denna implementering istället förändrar en geometri innefattar att det finns möjlighet att kombinera till exempel adjointmetoden och surrogatmodellen. Det skulle vara intressant att först optimera en geometri med surrogatmodellering och sedan använda adjointmetoden för att ytterligare optimera vingen och dess geometri.

6

Slutsatser

Det huvudsakliga syftet med det här kandidatarbetet har varit att undersöka för- och nackdelar med tre olika optimeringsmetoder då de används för att aerodynamiskt formoptimera en bakvinge på en generisk GT3-bil. Målet har varit att försöka bibehålla negativ lyftkraft och sänka luftmotståndet jämfört med ett referensfall, se 3.1.1. Ett delmål har även varit att i slutet av arbetet kunna dra slutsatser om hur de olika optimeringsmetoderna förhåller sig till varandra avseende beräkningsmässig kostnad och resultat. Från framtagna resultat i form av simuleringar och figurer har slutsatser kunnat dras.

Surrogatmodellen är en interpolationsmetod som grundar sig i att dra lärdom av gamla genererade resultat, med given indata, för att sedan föreslå ny indata som förutses förbättra nya resultat. Denna metod har visat sig vara den bäst lämpade för detta projekt. Detta för att optimeringsvariablerna snabbt konvergerat mot specifika positioner och vinklar. Samtidigt har residualerna från de genererade lösningarna konvergerat mot ett såpass lågt värde att konvergens antas vara uppfyllt för respektive simulering. Värdet på lyftkraftskoefficienten, C_L , har ansatts till ett för hårt krav och därmed har inte luftmotståndskoefficienten, C_D , kunnat anta speciellt effektiva värden.

En genetisk algoritm grundar sig i att många värden på parametrar testas för att sedan plocka ut de som är bäst lämpade. Detta tillvägagångsätt är inte fördelaktigt med CFD inom fordonsaerodynamik på grund av den stora mängd beräkningar som krävs för att testa varje enskilt värde på de olika parametrarna.

Även om den genetiska algoritmen inte är rimlig att applicera inom fordonsaerodynamik på grund av mängden core-timmar som krävs, är den så pass lätt att applicera att det är fullt rimligt att använda vid mindre krävande simuleringar. Ett krav är då att algoritmen är helt automatisk, eftersom det är väldigt tidskrävande att påbörja nya simuleringar för hand.

Adjointmetoden är gradientbaserad optimeringsstrategi som i denna tillämpning optimerar formen på en geometri istället för att optimera utifrån ett antal parametrar. Jämfört med de andra optimeringsstrategierna är detta en unik metod. För att få en klar bild av adjointmetodens potential och möjligheter behöver fler iterationssteg köras. Inom ramarna för detta kandidatarbete fanns inga möjligheter att köra fler iterationssteg eftersom svårigheter med adjointlösarens konvergens uppstod i ett sent stadie av arbetet. Även en studie på vilket sätt konvergens för adjointlösaren kan uppnås behöver utföras för att tillämpa adjointmetoden på bästa sätt.

Sammanfattningsvis är adjointmetoden en optimeringsstrategi som inte är anpassad för att tillämpas på ett problem som i detta arbete. Adjointlösaren kräver inställningar för lösaren utöver grundinställningarna vilket ökar antalet beräkningar som behöver utföras. Att på detta sätt hjälpa adjointlösaren att konvergera ställer höga krav på den beräkningsmässiga tiden och kostnaden. Den beräkningsmässiga kostnaden är helt beroende av hur många optimeringscykler som ska utföras. Adjointmetoden tros dock ge bra resultat om den kombineras med ytterligare en optimeringsmetod såsom surrogatmodellering eller möjligtvis genetiska algoritmer. Då kan den andra optimeringsmetoden användas i det tidiga optimeringsstadiet och adjointmetoden i slutskedet av optimeringsprocessen. Slutligen rekommenderas det att använda sig av samma krav på C_L för respektive optimeringsmetod för att på ett bättre sätt kunna jämföra dessa.

Litteraturförteckning

- [1] W. F. Milliken och D. L. Milliken. *Race Car Vehicle Dynamics*. USA, 1995. ISBN: 978-1-56091-526-3.
- [2] F. White. *Fluid Mechanics*. 8th ed. USA, 2016.
- [3] *DrivAer Model*. TUM – Technische Universität München. URL: <http://www.aer.mw.tum.de/en/research-groups/automotive/drivaer/> (hämtad 2019-03-07).
- [4] R. Eymard, G. Thierry och R. Herbin. “Finite Volume Methods”. I: *Handbook of Numerical Analysis* 7 (31 dec. 2000), s. 713–1018. DOI: 10.1016/S1570-8659(00)07005-8.
- [5] *Finite volume*. CFD Online. URL: https://www.cfd-online.com/Wiki/Finite_volume (hämtad 2019-04-30).
- [6] F. R. Menter. “Two-equation eddy-viscosity turbulence models for engineering applications”. I: *AIAA Journal* 32.8 (1 aug. 1994), s. 1598–1605. ISSN: 0001-1452. DOI: 10.2514/3.12149. URL: <https://arc.aiaa.org/doi/10.2514/3.12149> (hämtad 2019-05-11).
- [7] S. Hoyas och J. Jimenez. *DNS data of fully developed plane turbulent channel flow*. URL: <https://torroja.dmt.upm.es/ftp/channels/data/statistics/> (hämtad 2019-05-12).
- [8] *CO2 emissions and fuel consumption: what is the link?* WLTPfacts.eu. URL: <https://wltfacts.eu/link-between-co2-emissions-fuel-consumption/> (hämtad 2019-05-06).
- [9] G. Fontaras, N.-G. Zacharof och B. Ciuffo. “Fuel consumption and CO2 emissions from passenger cars in Europe – Laboratory versus real-world emissions”. I: *Progress in Energy and Combustion Science* 60 (maj 2017), s. 97–131. ISSN: 0360-1285. DOI: 10.1016/j.pecs.2016.12.004. URL: <http://www.sciencedirect.com/science/article/pii/S0360128516300442> (hämtad 2019-05-06).
- [10] *Mänsklig påverkan*. Världsnaturfonden WWF. URL: <https://www.wwf.se/klimat/mansklig-paverkan/> (hämtad 2019-05-06).

-
- [11] Simon McBeath. “Chapter 5, Wings”. I: *Competition Car Aerodynamics*. 1st ed. Sparkford: Haynes Publishing, 2006, s. 81–101.
- [12] T. Braconnier m. fl. “Towards an adaptive POD/SVD surrogate model for aeronautic design”. I: *Computers & Fluids* 40.1 (1 jan. 2011), s. 195–209. ISSN: 0045-7930. DOI: 10.1016/j.compfluid.2010.09.002. URL: <http://www.sciencedirect.com/science/article/pii/S0045793010002306> (hämtad 2019-03-26).
- [13] A I. J. Forrester, A. Sóbester och A. J. Keane. *Engineering design via surrogate modelling: a practical guide*. Wiley, 2008. 228 s. ISBN: 978-0-470-06068-1.
- [14] R. L. Hardy. “Multiquadric equations of topography and other irregular surfaces”. I: *Journal of Geophysical Research (1896-1977)* 76.8 (10 mars 1971), s. 1905–1915. ISSN: 2156-2202. DOI: 10.1029/JB076i008p01905. URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/JB076i008p01905> (hämtad 2019-04-08).
- [15] B. Fornberg och C. Piret. “On choosing a radial basis function and a shape parameter when solving a convective PDE on a sphere”. I: *Journal of Computational Physics* 227.5 (20 febr. 2008), s. 2758–2780. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2007.11.016. URL: <http://www.sciencedirect.com/science/article/pii/S0021999107005025> (hämtad 2019-04-10).
- [16] M. Wahde. *Biologically inspired optimization methods*. WIT Press, 2008. 218 s. ISBN: 978-1-84564-148-1.
- [17] C. Othmer. “Adjoint methods for car aerodynamics”. I: *Journal of Mathematics in Industry* 4.1 (2014), s. 6. ISSN: 2190-5983. DOI: 10.1186/2190-5983-4-6. URL: <http://mathematicsinindustry.springeropen.com/articles/10.1186/2190-5983-4-6> (hämtad 2019-03-07).
- [18] JiaQi Luo, JunTao Xiong och Feng Liu. “Aerodynamic design optimization by using a continuous adjoint method”. I: *Science China Physics, Mechanics & Astronomy* 57.7 (juli 2014), s. 1363–1375. ISSN: 1869-1927. DOI: 10.1007/s11433-014-5479-0. URL: <https://doi.org/10.1007/s11433-014-5479-0> (hämtad 2019-04-30).
- [19] J. Brezillon och N. R. Gauger. “2D and 3D aerodynamic shape optimisation using the adjoint approach”. I: *Aerospace Science and Technology* 8.8 (dec. 2004), s. 715–727. ISSN: 1270-9638. DOI: 10.1016/j.ast.2004.07.006. URL: <http://www.sciencedirect.com/science/article/pii/S1270963804000914> (hämtad 2019-04-30).
- [20] *S1223 (s1223-il)*. Airfoil Tools. URL: <http://airfoiltools.com/airfoil/details?airfoil=s1223-il> (hämtad 2019-05-02).

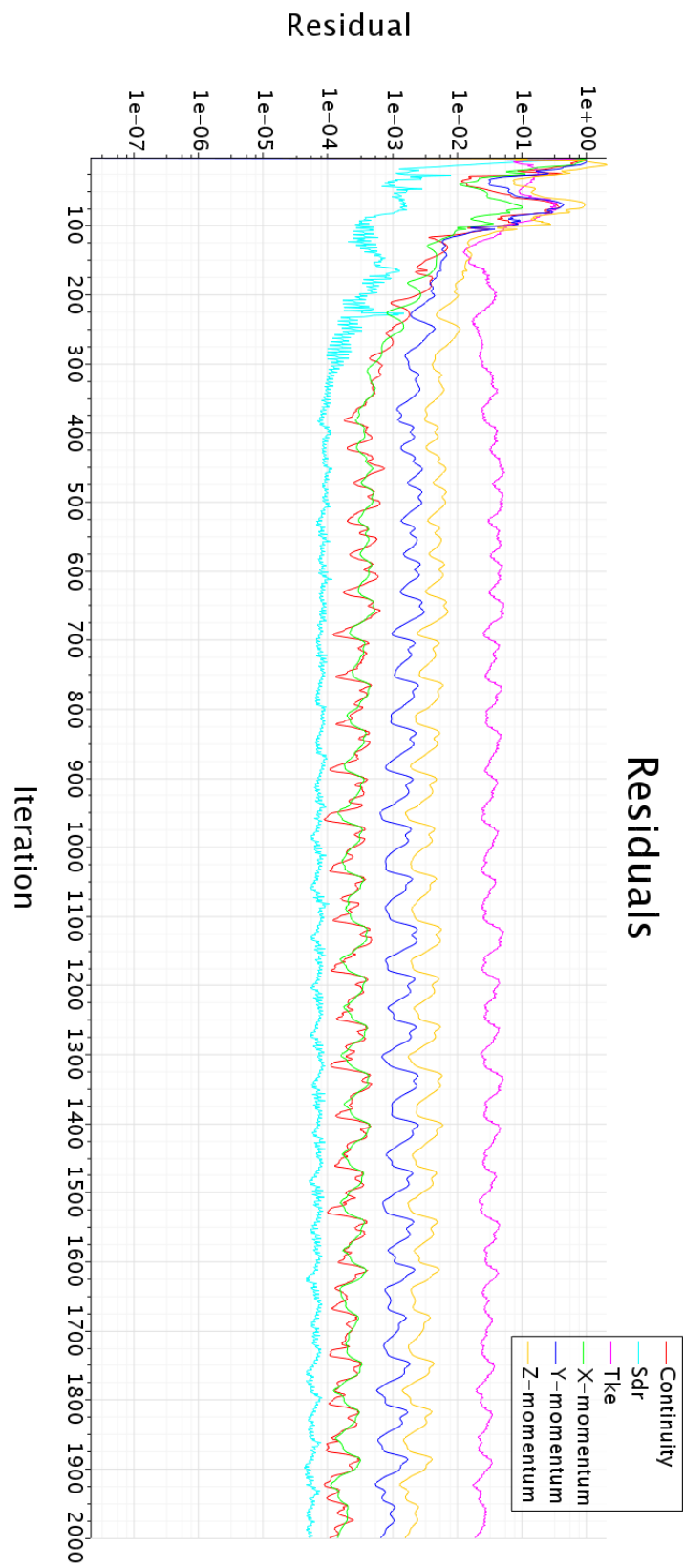
- [21] M. Kuron. *3 Criteria for Assessing CFD Convergence*. engineering.com. 6 jan. 2015. URL: https://www.engineering.com/DesignSoftware/DesignSoftwareArticles/ArticleID/9296/3-Criteria-for-Assessing-CFD-Convergence.aspx?fbclid=IwAR2q0XFVhZ-hhmGuj3f5V44h0QgW_weCEFEu_AxQHZmprre1IUnCukVZquQ (hämtad 2019-05-06).
- [22] *Adjoint Solver Properties*. The Steve Portal. URL: https://documentation.thesteveportal.plm.automation.siemens.com/starccmplus_latest_en/index.html?param=1RYSu&authLoc=https://thesteveportal.plm.automation.siemens.com/AuthoriseRedirect#page/STARCCMP%2FGUID-7B87A2C7-840A-4785-B0BA-B0D8005E8A6F%3Den%3D.html (hämtad 2019-05-12).
- [23] R. Ahmad m.fl. *Minskat luftmotstånd på bilar genom automatisk formoptimering*. 2017. URL: <http://studentarbeten.chalmers.se/publication/250459-minskat-luftmotstand-pa-bilar-genom-automatisk-formoptimering>.
- [24] *Adjoint Analysis Workflow*. The Steve Portal. URL: https://documentation.thesteveportal.plm.automation.siemens.com/starccmplus_latest_en/index.html?param=1RYSu&authLoc=https://thesteveportal.plm.automation.siemens.com/AuthoriseRedirect#page/STARCCMP%2FGUID-90A1ED16-6EBF-444C-BDEF-7A90F55F7B0D%3Den%3D.html (hämtad 2019-05-12).

B

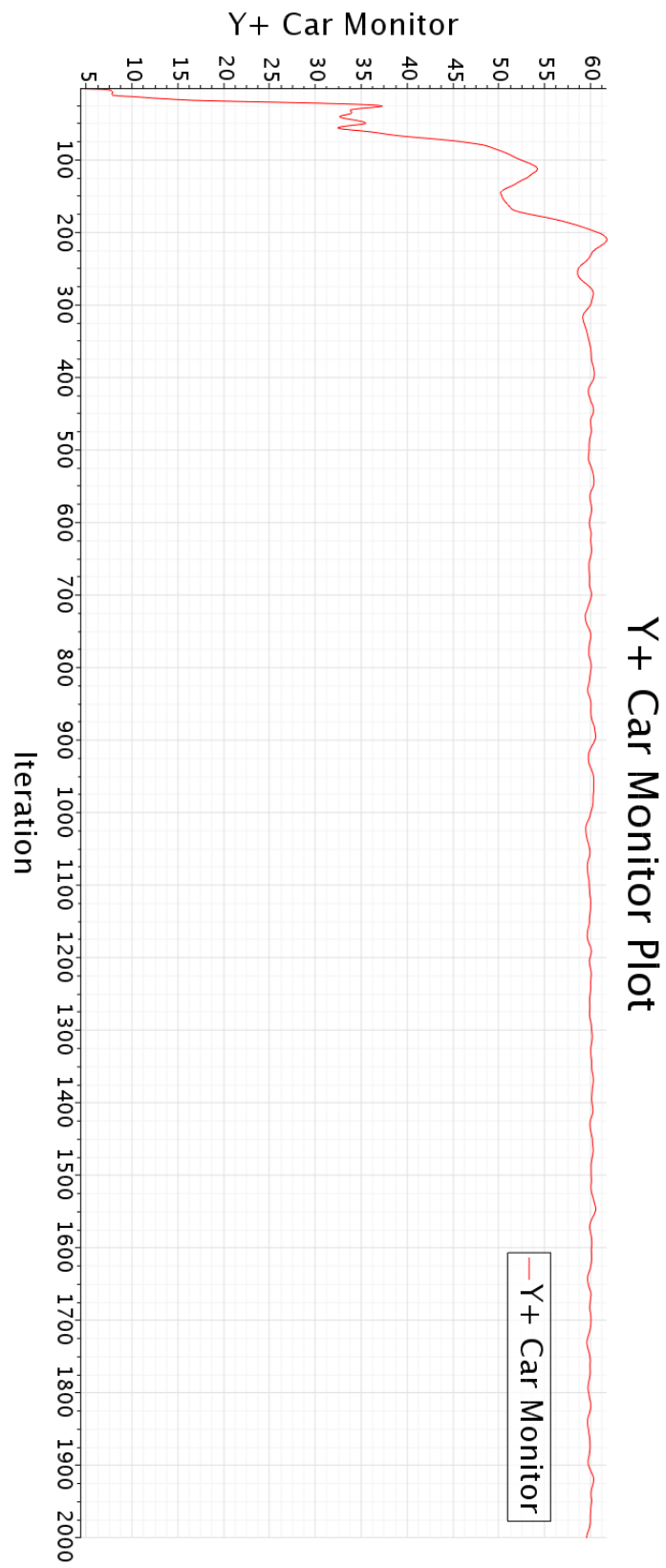
Surrogatmodellen

Tabell B.1: Alla simuleringar för surrogatmodellen.

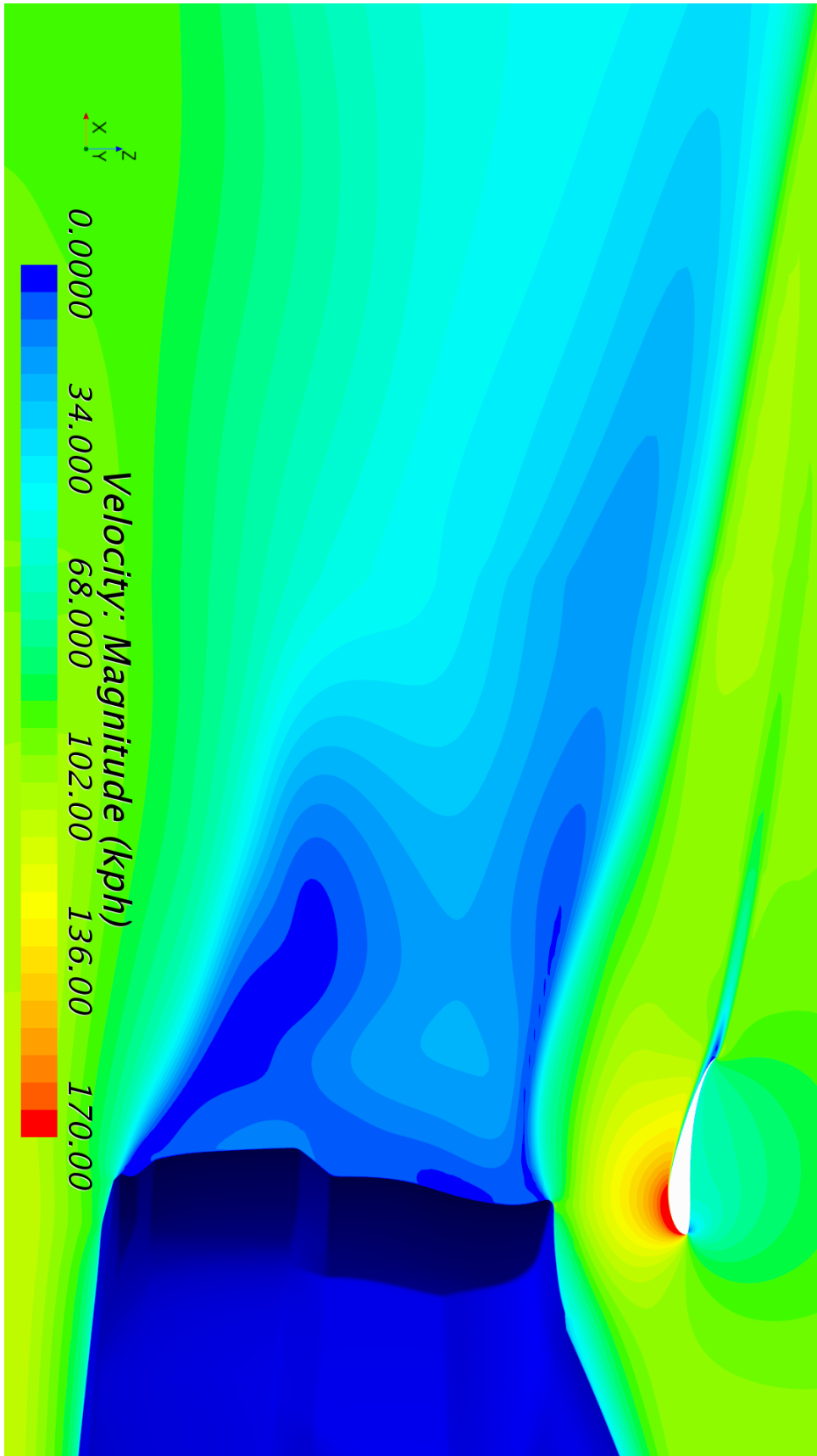
Simulering	Höjd [m]	Vinkel [°]	Horisontellt [m]	Typ	C_D	C_L
1	0,3256	27,15	0	LHS	0,2635	-0,2869
2	0,2792	63,98	0	LHS	0,4855	-0,5267
3	0,4691	5,97	0	LHS	0,2377	-0,0443
4	0,2287	41,19	0	LHS	0,3473	-0,4238
5	0,4338	56,99	0	LHS	0,4057	-0,4526
6	0,4999	69,99	0	STD	0,4578	-0,4845
7	0,3187	43,04	0	MEDIAN	0,3378	-0,3905
8	0,2000	70,00	0	STD	0,4412	-0,4769
9	0,2000	42,94	0	MEDIAN	0,3969	-0,4852
10	0,3915	65,00	0	STD	0,4649	-0,5042
11	0,3514	44,79	0	MEDIAN	0,3432	-0,3933
12	0,2000	40,64	0	MEDIAN	0,3687	-0,4571
13	0,2525	61,92	0	STD	0,4650	-0,4835
14	0,2066	37,50	0	MEDIAN	0,3314	-0,4111
15	0,2000	53,68	0	STD	0,4300	-0,4611
16	0,2049	38,53	0	MEDIAN	0,3456	-0,4272
17	0,1000	39,99	0	STD	0,3266	-0,3264
18	0,1000	35,75	0	MEDIAN	0,3100	-0,3220
19	0,1787	39,96	0	STD	0,3636	-0,4424
20	0,1998	38,36	0	MEDIAN	0,3461	-0,4368
21	0,1659	40,00	0	STD	0,3836	-0,4622
22	0,1992	38,11	0	MEDIAN	0,3446	-0,4315
23	0,2260	26,82	0,0831	STD	0,2812	-0,3944
24	0,1420	29,57	-0,0074	RAND	0,3074	-0,4036
25	0,1556	40,00	0,15	STD	0,3763	-0,3746
26	0,1878	34,00	0,15	MEDIAN	0,3042	-0,3766
27	0,1672	36,80	-0,0178	STD	0,3383	-0,4180
28	0,2467	39,65	0,0021	STD	0,3284	-0,3993
29	0,2276	37,77	0,0005	MEDIAN	0,3157	-0,3865
30	0,1563	39,79	0,0151	STD	0,3740	-0,4199
31	0,1950	37,65	0,0127	MEDIAN	0,3356	-0,4178
32	0,1627	38,18	-0,0183	STD	0,3580	-0,4435
33	0,1948	37,68	-0,0078	MEDIAN	0,3382	-0,4155
34	0,2489	26,97	0,1296	STD	0,2798	-0,3985
35	0,2028	35,97	0,0673	RAND	0,3087	-0,3778
36	0,1799	39,41	0,0185	STD	0,3583	-0,4489
37	0,1866	37,58	-0,0104	MEDIAN	0,3444	-0,4281
38	0,1804	37,19	-0,0267	MEDIAN	0,3444	-0,4247
39	0,2129	39,19	-0,0205	MEDIAN	0,3378	-0,4202
40	0,2027	38,12	0,0036	MEDIAN	0,3448	-0,4283



Figur B.1: Residualer för simulering 39 med surrogatmodellern.



Figur B.2: y^+ på bilen för simulering 39 med surrogatmodellen.



Figur B.3: Hastighetsprofil för simulering nummer 34 med annat krav på C_L .

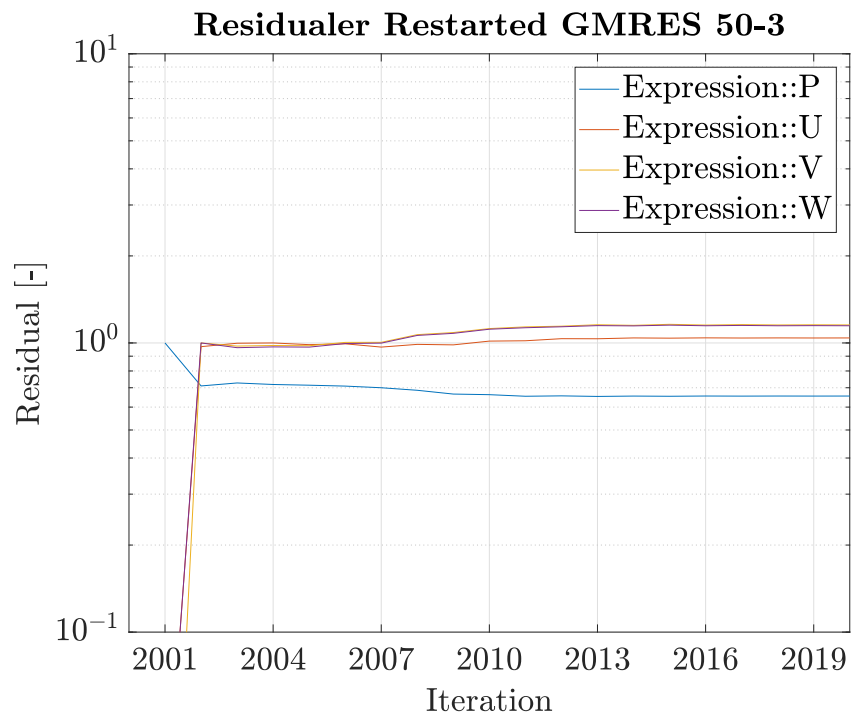
C

Genetiska algoritmen

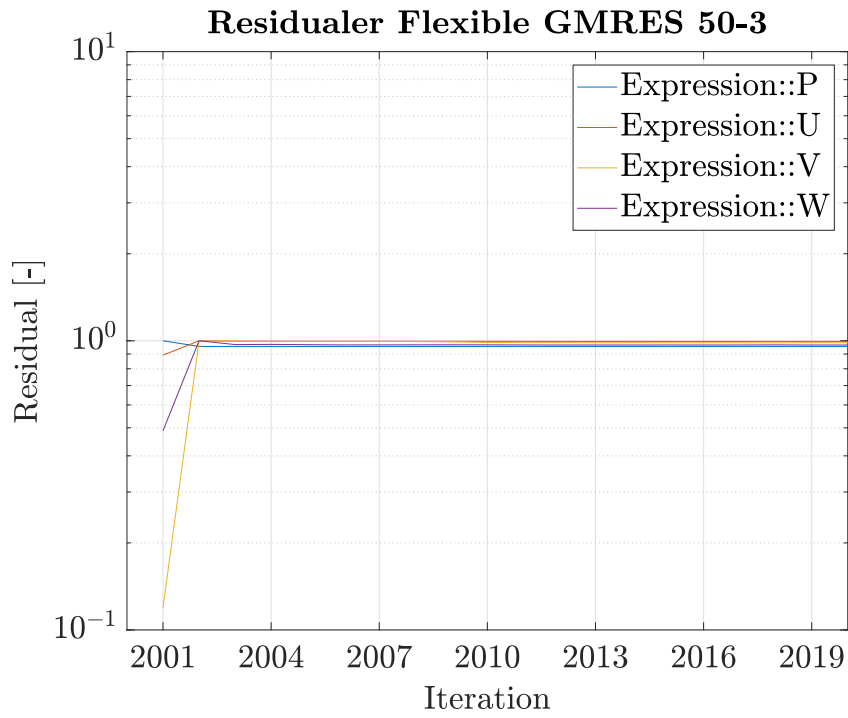
Filnamn	Vinkel [°]	Höjd [cm]	C_L	C_D	Körs
bil.sim	-	-	0,0049	0,2291	
Generation 1					
surro1.sim	27,15	32,56	-0,2869	0,2635	Y
surro2.sim	63,98	27,92	-0,5267	0,4855	Y
surro3.sim	5,97	46,91	-0,0443	0,2377	Y
surro4.sim	41,19	22,87	-0,4238	0,3473	Y
surro5.sim	56,99	43,38	-0,4526	0,4057	Y
Generation 2					
G2i1.sim	40,00	11,04	-0,3362	0,3373	Y
G2i2v3.sim	40,00	46,90	-0,2929	0,3047	Y
G2i3.sim	40,00	43,38	-0,3022	0,3074	Y
G2i4.sim	37,33	15,54	-0,4225	0,3411	Y
G2i5.sim	12,27	28,77	-0,2147	0,2349	Y
Generation 3					
G3i1.sim	14,35	28,77	-0,2381	0,2385	Y
G3i2.sim	12,27	43,38	-0,1943	0,2390	Y
G3i3.sim	26,03	11,04	-0,3549	0,2984	Y
G3i4.sim	40,00	28,77	-0,3797	0,3211	Y
G2i5.sim	12,27	28,77	-0,2147	0,2349	N(G2i5)
Generation 4					
G4i1.sim	26,03	11,04	-0,3549	0,2984	N(G3i3)
G4i2.sim	22,14	11,64	-0,3536	0,2835	Y
G4i3.sim	12,27	28,77	-0,2147	0,2349	N(G2i5)
G4i4.sim	26,03	28,77	-0,3396	0,2657	Y
G5i5.sim	26,03	28,77	-0,3396	0,2657	N(G4i4)
Generation 5					
G5i1.sim	22,25	22,25	-0,3648	0,2755	Y
G5i2.sim	26,03	11,04	-0,3549	0,2984	N(G3i3)
G5i3.sim	26,03	28,77	-0,3396	0,2657	N(G4i4)
G5i4.sim	22,14	11,04	-0,3536	0,2868	Y
G5i5.sim	22,14	11,04	-0,3536	0,2868	N(G5i4)
Generation 6					
G6i1.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)
G6i2.sim	26,03	12,34	-0,3834	0,3006	Y
G6i3.sim	35,43	11,04	-0,3437	0,3229	Y
G6i4.sim	22,14	11,04	-0,3536	0,2826	N(G5i4)
G6i5.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)
Generation 7					
G7i1.sim	27,08	22,25	-0,3309	0,2739	Y
G7i2.sim	26,03	11,04	-0,3549	0,2984	N (G5i2)
G7i3.sim	26,03	22,25	-0,3648	0,2755	N (G5i1)
G7i4.sim	26,03	22,25	-0,3648	0,2755	N (G5i1)
G7i5.sim	26,03	22,25	-0,3648	0,2755	N (G5i1)
Generation 8					
G8i1.sim	25,98	22,85	-0,3595	0,2744	Y
G8i2.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)
G8i3.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)
G8i4.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)
G8i5.sim	26,03	22,25	-0,3648	0,2755	N(G5i1)

D

Adjointmethoden



Figur D.1: Residualer för *Restarted GMRES* 50-3.



Figur D.2: Residualer för *Flexible GMRES* 50-3.



Figur D.3: Residualer för *Flexible GMRES* 100-10 och diskretisering på *2nd-order*.