

Echtzeit-Strömungssimulationen auf Grafikkarten in der automobilen Produktentwicklung

Christian F. Janßen¹, Thorsten Grahs^{2,3}

¹ Institut für Fluidodynamik und Schiffstheorie, Technische Universität Hamburg-Harburg

² Institut für Wissenschaftliches Rechnen, Technische Universität Braunschweig

³ move-csc, Methodenentwicklung, Braunschweig (<http://www.move-csc.de>)

Die Nachfrage nach schnellen und gleichzeitig verlässlichen numerischen Simulationen für komplexe Fragestellungen in Forschung und Entwicklung in der automobilen Industrie hat in den vergangenen Jahren stetig zugenommen. In gleichem Maße sind die verfügbaren Hardware-Ressourcen gestiegen und die Preise für Hochleistungs-Hardware gesunken. Dennoch existieren weiterhin zahlreiche komplexe Fragestellungen, deren numerische Simulation in Hinblick auf Rechenzeiten und verfügbare Ressourcen weiterhin zu anspruchsvoll ist, um direkt in einen Produktentwicklungszyklus eingebettet zu werden.

In diesem Beitrag stellen wir eine Machbarkeitsstudie zur direkten Integration hochentwickelter, optimierter CFD-Verfahren in den automobilen Entwicklungsprozess vor. Um kurze Simulationszyklen sicherzustellen, werden besonders effiziente numerische Simulationswerkzeuge in Kombination mit spezieller Grafikkarten-Hardware eingesetzt. Der verwendete Löser *elbe* ist eine Simulationsumgebung zur numerischen Simulation zwei- und dreidimensionaler Strömungen mit freien Oberflächen und basiert auf einem innovativen Gitter-Boltzmann-Verfahren.

Die numerische Methode wird anhand der Simulation der Wasserdurchfahrt eines Serienfahrzeuges untersucht. Der gewählte Simulationstestfall ist dabei beabsichtigt anspruchsvoll: einerseits ist das transiente, nicht-lineare Strömungsverhalten inklusive der Bewegung der freien Wasseroberfläche zu simulieren, andererseits erfordert die sich bewegende komplexe dreidimensionale Fahrzeuggeometrie eine gleichzeitige Gitteranpassung in jedem Zeitschritt der numerischen Berechnung. Entsprechende numerische Simulationen mit klassischen Finite-Volumen-Verfahren auf HPC-Clustern mittlerer Größe benötigen einige Tage oder gar Wochen Rechenzeit. Die *elbe*-Umgebung, in Kombination mit aktueller Hochleistungs-GPU-Hardware von NVIDIA, ermöglicht vergleichbare Simulationen innerhalb weniger Stunden, auf einer einzelnen Grafikkarte in einer gewöhnlichen Workstation.

1 Einleitung

Die rasante Zunahme der verfügbaren Rechenleistungen im Bereich des Hochleistungsrechnens (engl. high performance computing, HPC) hat insbesondere in Kombination mit der drastischen Kostensenkung der Hardware zu einer starken Nachfrage nach numerischen Simulationen und deren Integration in die Entwicklungsprozesse geführt. Dabei führt die hohe Komplexität der realen technischen Anwendungen jedoch unverändert zu hohen Rechen- bzw. Antwortzeiten der Simulationsumgebung, wodurch sich die Integration der Simulationen in die Prozesskette der Produktentwicklung schwierig gestaltet. Die Simulationsergebnisse sind üblicherweise erst nach Tagen oder Wochen verfügbar, während eine nahtlose Integration in die schnelllebige Produktentwicklung von bspw. Fahrzeugen Simulationsantwortzeiten von einigen Stunden bis hin zu maximal einem Tag erfordert. Dem gegenüber steht eine neue Entwicklung im HPC-Bereich: das wissenschaftliche Rechnen auf Grafikkarten. Ursprünglich entwickelt für die massiv-parallelen Bedürfnisse von hochauflösenden Videospielen, entstanden daraus leistungsstarke grafische Verarbeitungseinheiten, die für den massiven Durchsatz von numerischen Daten maßgeschneidert sind. Hierbei spielt es keine Rolle, ob Grafikdaten (Pixel/Texturen) oder Simulationsdaten verarbeitet werden sollen. Erste bahnbrechende Arbeiten im CFD-Bereich nutzten dabei direkt die Rendering-Pipeline und versteckten die eigentlichen numerischen Operationen in Visualisierungs-Anweisungen. Durch die Bereitstellung von Software Development Kits (SDKs) durch die Grafikkartenhersteller und eine zunehmende Anpassungen der Hardware-Architekturen ist die Programmierung jedoch nach und nach einfacher geworden. Das aktuelle NVIDIA Flaggschiff, die in Abb. 1 (links) dargestellte NVIDIA Tesla K40 Grafikkarte, ist speziell für wissenschaftliches Rechnen optimiert worden, bietet insgesamt 2880 Rechenkerne sowie 12 GB Grafikkar-

tenspeicher und verzichtet gänzlich auf die Möglichkeit, einen Bildschirm anzuschließen. Geblieben ist der durch die Massenproduktion bedingte relativ geringe Preis von etwa 3000€ sowie der einfache Zugriff auf die Rechenressourcen, da solche Karten in nahezu jedem Standard-PC mit PCI-Express-Schnittstelle installiert werden können. Werden wie in Abb. 1 (rechts) dargestellt mehrere Karten zu einem kleinen Multi-GPU-Cluster zusammengeschaltet, erhöht sich dementsprechend die Rechenleistung. Eine ideale Basis für lokale Simulationen und *Supercomputing on the Desktop*.



Abb. 1: NVIDIA Tesla K40 GPU (links). Multi-GPU Server mit 4 NVIDIA Tesla C2075 GPUs (rechts).

Der in diesem Artikel untersuchte Ansatz zur Integration komplexer Multi-Physik-Anwendungen in den Produktentwicklungszyklus kombiniert die hohe Rechenleistung von durchsatzorientierten GPUs mit einem speziell für diese Hardware geeigneten numerischen Verfahren, der Gitter-Boltzmann-Methode (engl. Lattice Boltzmann Method, LBM). Der Algorithmus ist aufgrund seiner lokalen Natur besonders gut auf Vektor-Prozessoren und Grafikkarten-Hardware einsetzbar. Im Vergleich zu Standard-CFD-Lösern auf Basis von Finite-Volumen-Methoden (FVM), lassen sich Rechenzeitvorteile von bis zu einem Faktor 5 bis 10 (abhängig von der Anwendung und den Implementierungsdetails der jeweiligen Codes) feststellen.

2 elbe – eine effiziente Gitter-Boltzmann Entwicklungsumgebung

Der numerische Löser elbe [1] ist ein effizientes Werkzeug zur numerischen Simulation komplexer zwei- und dreidimensionaler Strömungsprobleme. Der Löser berücksichtigt nicht-lineare Strömungsphänomene sowie den Einfluss freier Oberflächen, Viskosität und Turbulenz. elbe basiert auf einem Gitter-Boltzmann-Verfahren auf äquidistanten kartesischen Gittern und ist von zweiter Ordnung genau in Raum und Zeit. Die Methode ist explizit in der Zeit und erfordert lediglich Interaktionen zwischen direkt benachbarten Gitterknoten, so dass die Methode besonders effizient im Kontext aktueller Hochleistungshardware wie beispielsweise Grafikkarten anwendbar ist. Die Wahl einer effizienten Methode wie der LBM ermöglicht dabei bereits einige numerische Simulationen in Echtzeit. Die in Abbildung 2 dargestellte Innenraumströmung in einem A380 Cabin-Mockup ließ sich trotz einer Gitterknotenanzahl von 2,14 Millionen mit elbe bereits in Echtzeit simulieren und visualisieren. Die Berechnung von 120 Sekunden Strömungsverhalten benötigte dabei auf einer Grafikkarte in etwa 120 Sekunden Rechenzeit.

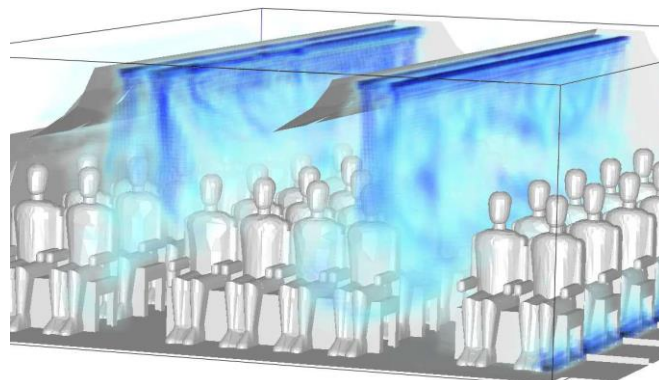


Abb. 2: Echtzeit-Simulation und –Visualisierung der Strömung in einem A380 Cabin Mockup mit elbe und elbeVIS. Simulationszeit 120 Sekunden, Rechenzeit etwa 120 Sekunden.

Gitter-Boltzmann-Verfahren gehen auf das stochastische Verhalten hypothetischer Gasteilchen zurück. Lösungen der Gitter-Boltzmann-Gleichung erfüllen die inkompressiblen Navier-Stokes-Gleichungen mit Diskretisierungsfehlern von $\mathcal{O}(\Delta x^2)$ und $\mathcal{O}(Ma^2)$. Da die in diesem Beitrag untersuchten Strömungen Reynoldszahlen im Bereich mehrerer Millionen aufweisen, wird ein Grobstrukturansatz zur Modellierung turbulenter Effekte verwendet. Zur Erfassung der freien Wasseroberfläche wurde in elbe aufgrund der Konservativität und Robustheit ein Volume-of-Fluid Ansatz gewählt, der die Oberfläche auf Basis der relativen Phasenvolumina in den Interfacezellen rekonstruiert. Das so aufgebaute und in elbe implementierte Verfahren wurde bereits auf zahlreiche Testfälle angewendet und erfolgreich validiert, bspw. für Szenarien mit freien Oberflächen und für Einphasenströmungen ohne freie Oberflächen (siehe Abb. 3).

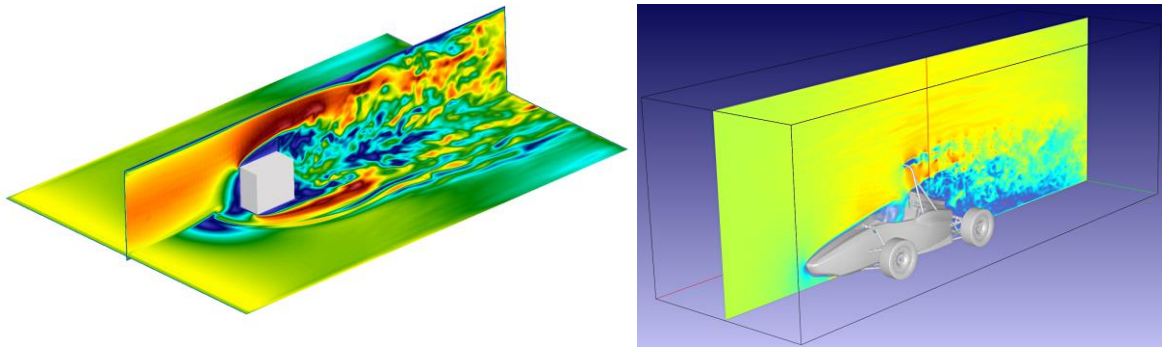


Abb. 3: Numerischer Windkanal in elbe. Validierung anhand klassischer Testfälle aus der Literatur (links). Anwendung am Beispiel des Formula Student Racers EGN2014 der TUHH (rechts).

3 Simulation einer Wasserdurchfahrt

Der vorgestellte Algorithmus wird nun auf die numerische Simulation der Wasserdurchfahrt eines PKW angewendet. Bedingt durch die stetig steigende Nachfrage nach SUVs und geländegängigen Fahrzeugen wächst die Bedeutung der Wasserdurchfahrts-Szenarien zur Beurteilung der Designqualität eines Fahrzeugentwurfs. Aus lizenztechnischen Gründen sind für diese Veröffentlichung vergleichbare Berechnungen auf Basis der DrivAer Geometrie [5] der TU München durchgeführt worden, die in Bezug auf Abmessungen, Form und Detailgrad mit einem realen Serienfahrzeug vergleichbar ist (Abbildung 4). Die DrivAer Geometrien sind frei verfügbar und wurden explizit als Validierungsmodelle für automobiler Applikationen erstellt.

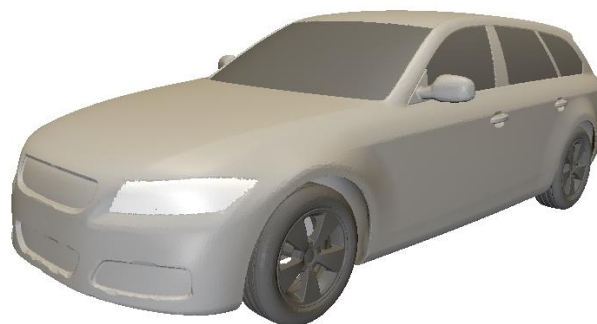


Abbildung 4: DrivAer Fahrzeuggeometrie [5]

Die Simulation wird für einen numerischen Wasserkanal einer Länge von 20 Metern, einer Breite von 5 Metern und einer Höhe von 3.75 Metern durchgeführt. Der Glatzwasserspiegel beträgt 0.4 Meter. Die Maschenweite des äquidistanten kartesischen Gitters beträgt $\Delta x = \Delta y = \Delta z = 0.025\text{m}$. Die Simulationszeit von 3 Sekunden wird mit einem Zeitschritt von $\Delta t = 0.0000325$ Sekunden diskretisiert. Die Fahrzeuggeschwindigkeit beträgt 4 m/s, was in etwa einer Froudezahl von 2 entspricht. Durchgeführt wurde die Berechnung auf einer NVIDIA Quadro 6000 Grafikkarte. Die Berechnungszeit für 3 Sekunden Strömungsverhalten auf dem oben skizzierten Gitter mit 24 Millionen Gitterzellen betrug lediglich 6 Stunden. Dies entspricht ungefähr 100 Millionen Knotenaktualisierungen je Sekunde (node updates per second, NUPS). In Abbildung 9 ist die Lage der freien Oberfläche für ausgewählte Zeitpunkte dar-

gestellt. Mit Beginn der Fahrzeugbewegung wird die freie Wasseroberfläche entsprechend ausgelenkt. Vor dem Fahrzeug bildet sich ein Wellenberg, hinter dem Fahrzeug erkennt man durch den Sog eine Absenkung der Wasseroberfläche. Aufgrund der fehlenden Berücksichtigung von Oberflächenspannungseffekten in dem verwendeten Modell bildet sich vor dem Wagen ein geschlossener Flüssigkeitsfilm anstelle der zu erwartenden Tröpfchenbildung. Dennoch entsprechen die Berechnungsergebnisse der Lage der freien Oberfläche in weiten Teilen den Referenz-Ergebnissen.

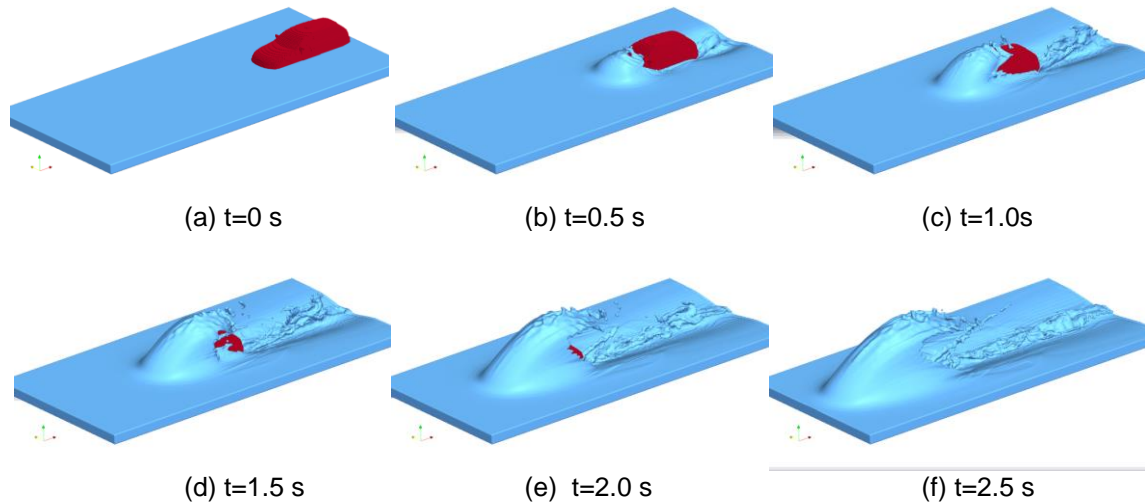


Abbildung 5: Zeitliche Entwicklung der Oberflächenauslenkung von $t=0\text{s}$ (a) bis $t=2.5\text{s}$ (f)

4 Zusammenfassung und Ausblick

In diesem Beitrag wurde die numerische Simulation der Wasserdurchfahrt eines Fahrzeugs vorgestellt, die im Rahmen einer Machbarkeitsstudie für einen Automobilhersteller untersucht wurde. Abgesehen von der Güte der numerischen Ergebnisse standen dabei die Rechenzeiten der numerischen Simulationen auf Grafik-Hardware im Vordergrund und insbesondere die Fragestellung ob es mittlerweile möglich ist, solche komplexen Ingenieur Anwendungen komplexer Strömungen mit freien Oberflächen und bewegter Geometrien in vertretbarer Rechenzeit zu adressieren.

Letzteres ist besonders für konventionelle CFD-Löser und FVM-Verfahren eine Herausforderung, da die Netzaktualisierung bzw. Bewegung zu einer hohen Anzahl Gitterzellen, kleinen Zeitschritten und hohen Rechenzeiten selbst auf CPU-Clustern führt. Im Gegensatz dazu kann elbe hier seine Vorteile ausspielen: der massiv-parallele Berechnungsansatz führt gemeinsam mit der Grafikhardware zu Rechenzeiten und Hardwarekosten, die nur ein Bruchteil des Ressourceneinsatzes konventioneller Löser betragen. Die Rechenzeiten des hier vorgestellten Falls (ca. 6 Stunden für ein Gitter mittlerer Auflösung) bestätigen das Potential des gewählten Verfahrens. Numerische Simulationen mit dem elbe-Verfahren ermöglichen somit, komplexe numerische Simulationen direkt in den Produkt-Entwicklungsprozess zu integrieren.

5 References

- [1] Janßen, C.: "elbe - efficient lattice boltzmann environment", <http://www.tuhh.de/elbe>
- [2] Janßen, C.: "Kinetic approaches for the simulation of non-linear free surface flow problems in civil and environmental engineering", PhD Thesis, TU Braunschweig, 2010.
- [3] Janßen, C. , Krafczyk, M.: "Free surface flow simulation on GPGPUs using the LBM", Computers and Mathematics with Applications, 2011, Bd. 61, 12.
- [4] Krafczyk, M., Tölke, J., Luo, L.-S.: "Large-eddy simulation with a multiple-relaxation-time LBE model", Int. J. Mod. Phys. 2003, Bd. B 17, S. 33-39.
- [5] Heft, A. I., Indinger, T. , Adams, N. A.: "Introduction of a New Realistic Generic Car Model for Aerodynamic", Michigan, USA, : Paper 2012-01-1068, 2012.