

Gastvorlesung

Concurrent Programming



Inhalt

- Über mich
 - Effiziente parallele Dosisberechnung 1. Teil 20'
 - Einführung in OpenCL 30'
 - Arbeitsblatt 30'
 - Effiziente parallele Dosisberechnung 1. Teil 10'
-

Über mich



- B. Sc. Computer Science
 - Thesis:
Effiziente parallele Dosisberechnung
 - Scala, Concurrent Programming, Machine Learning
-

Effiziente parallele Dosisberechnung

Concurrent Programming



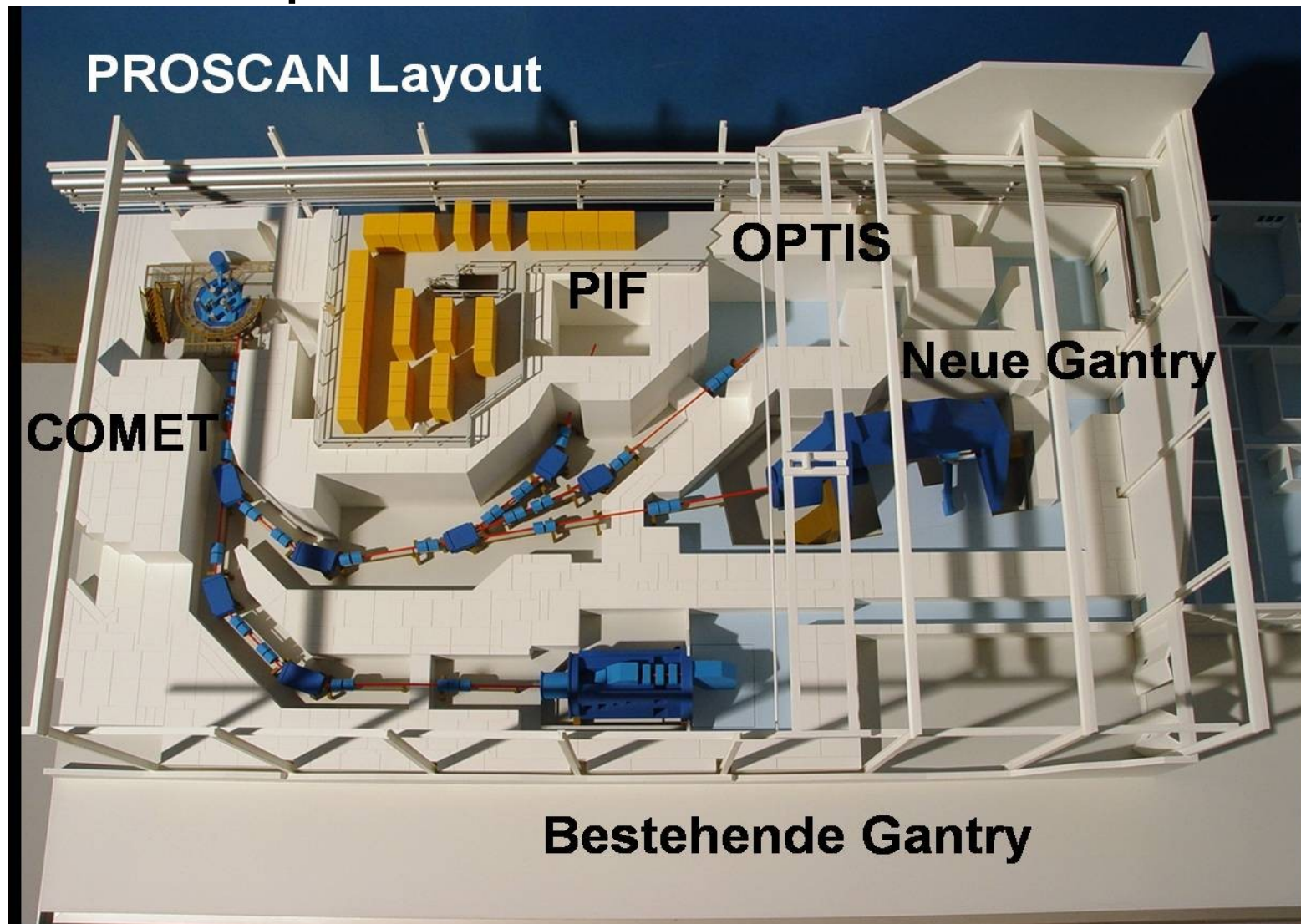
Inhalt

- Protonentherapie
 - Dosisberechnung
 - Resultate
 - Varianten
 - Performance
-

Protonentherapie am PSI

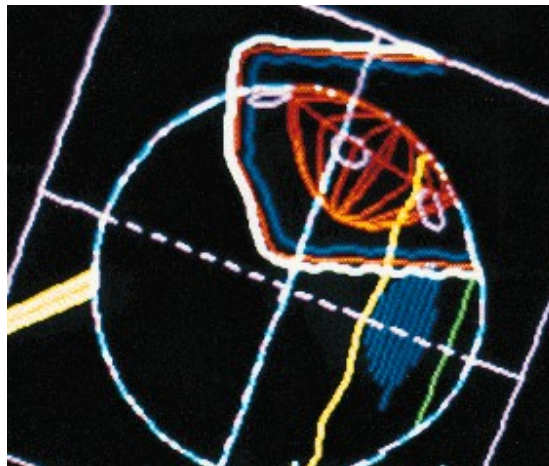
- Seit 1984
 - Über 5700 Patienten
 - Behandlung von
 - Augentumoren
 - Hirn-/Schädelbasis-/Wirbelsäulentumoren
 - (Prostatakarzinomen)
-

Protonentherapie am PSI



Protonentherapie am PSI – OPTIS 1 & 2

- Bestrahlung von Augentumoren



Protonentherapie am PSI – Gantry 1

- Bestrahlung von tiefliegenden Tumoren

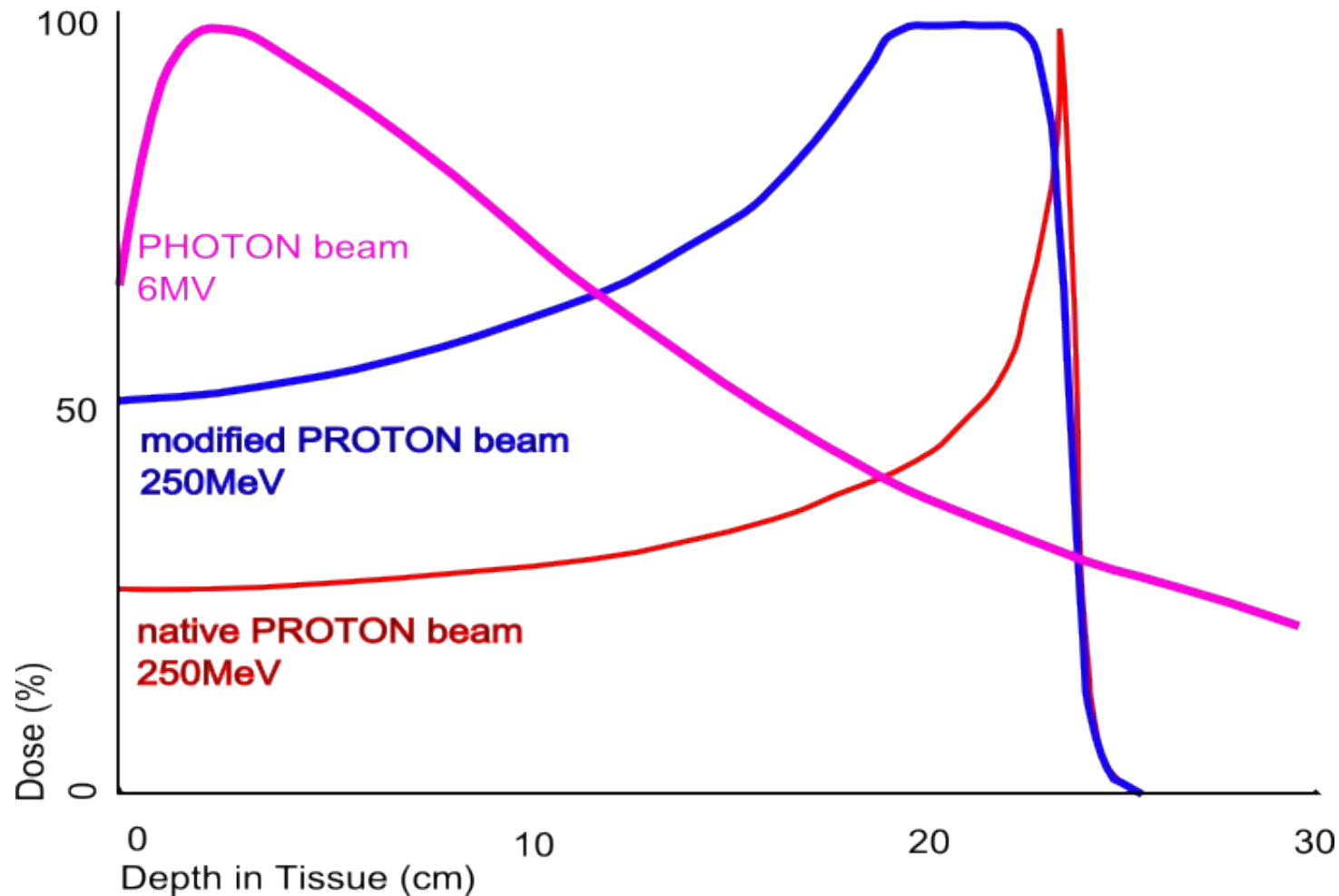


Protonentherapie am PSI – Gantry 2

- Bestrahlung von tiefliegenden Tumoren

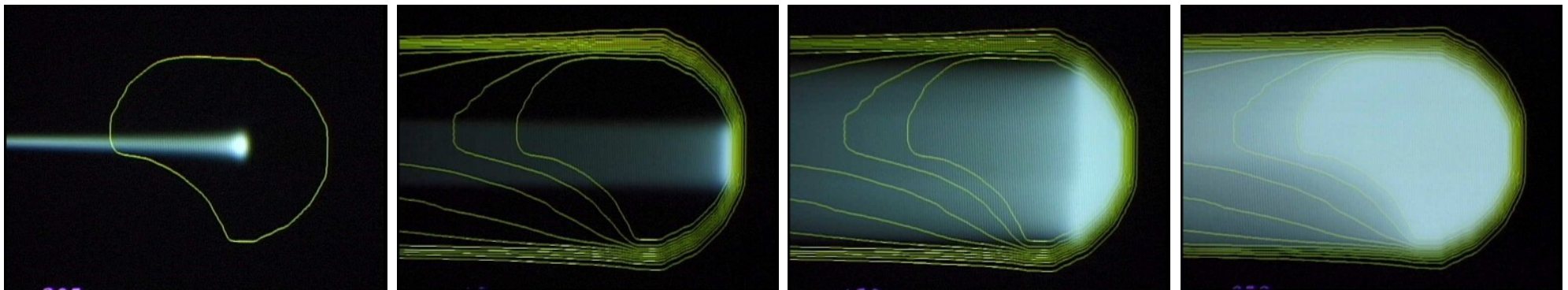


Weshalb Protonentherapie



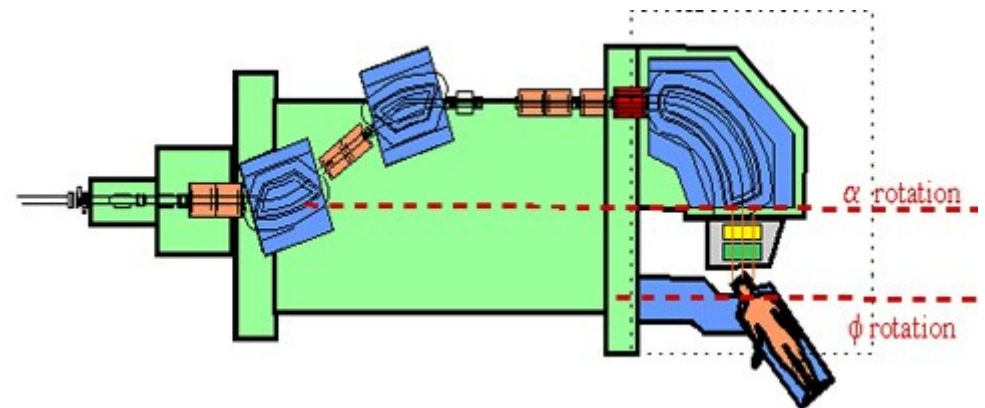
Dosisberechnung – Spot Scanning

- Berechnung mittels eines Pencil Beam Models
- Spot: Ein „Schuss“
- Scannen: Verschieben der Spots
- Abgelegte Dosis = Summe der Dosen der Spots



Dosisberechnung – Therapieplanung

- Arzt + Physiker planen Therapie
 - Identifizieren des Tumors
 - Definieren der Spots
- Optimierung der Freiheitsgrade der Gantry



Dosisberechnung – Ziele der Thesis

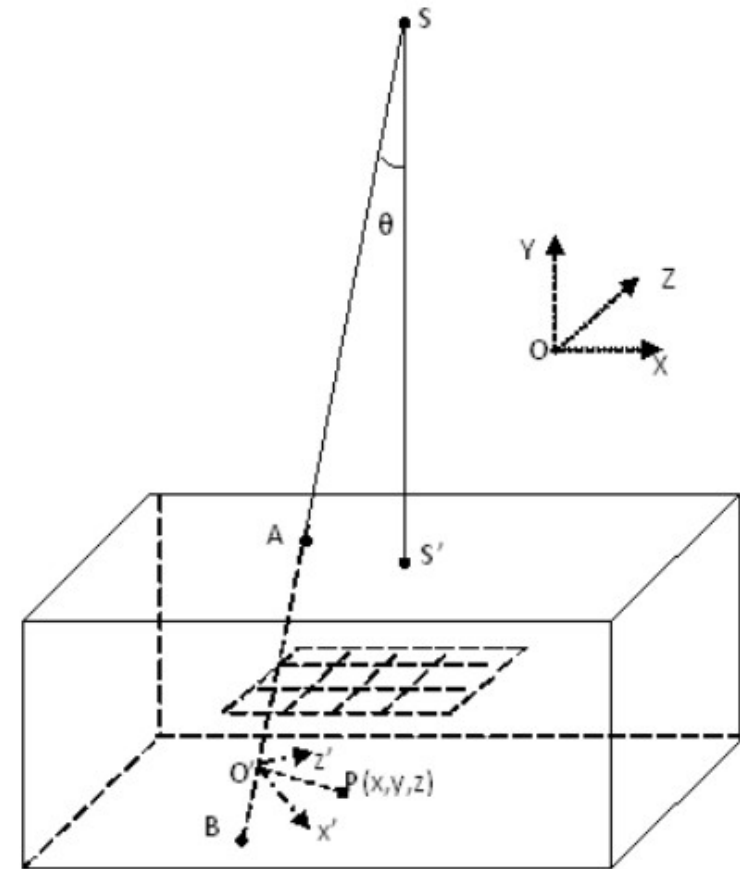
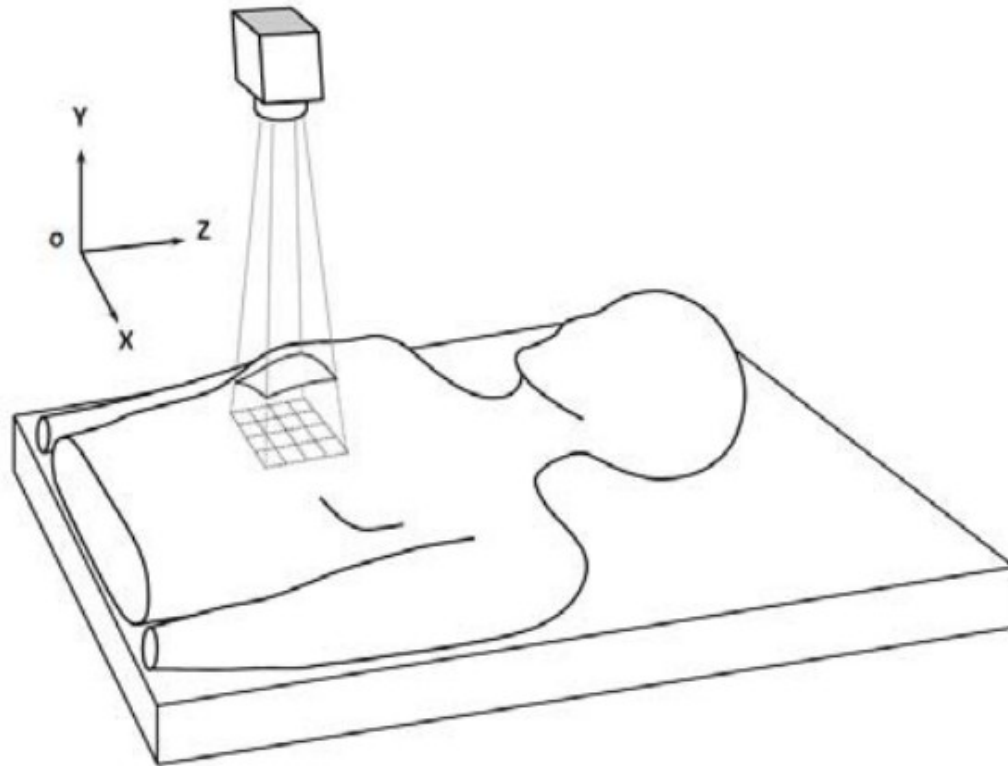
- So schnell wie möglich, so wartbar wie möglich
 - Evaluieren verschiedener Varianten
 - Möglichst auf der JVM
 - Empfehlungen welche Technologie weiterverfolgt werden soll
 - Verbesserungen des Algorithmus
-

Dosisberechnung – Performance?

- UI der Therapieplanungs-Software
- Optimierung
- Zukunft:
 - Online adaptive radiation therapy (OART)

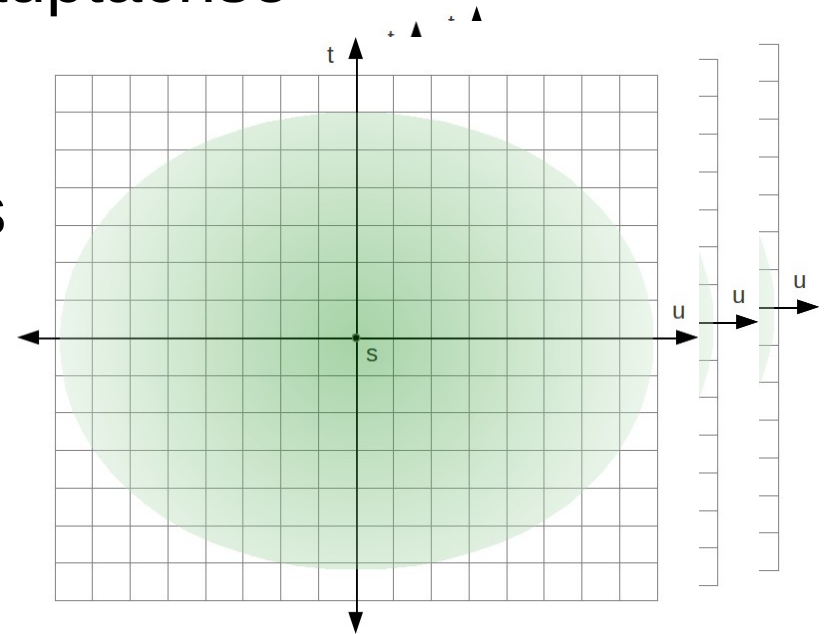


Dosisberechnung – Algorithmus



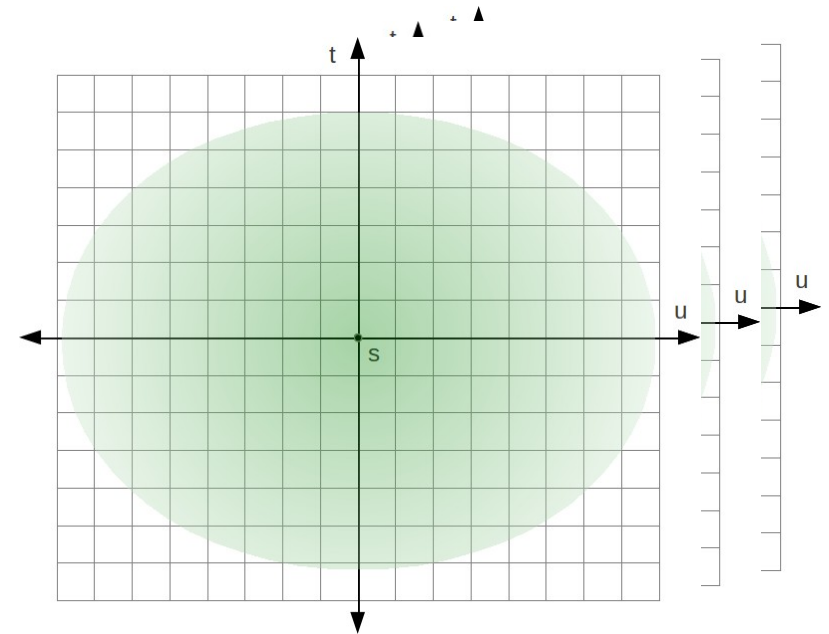
Dosisberechnung – Berechnung der Dosis

- Abhängig von der Water Equivalent Depth
- Abhängig vom Air Gap (Abstand Patient – Nozzle)
- Abhängig vom Abstand zur Spot-Hauptachse
- Optimierungen:
 - Abschneiden nachdem 99% Dosis deponiert wurde



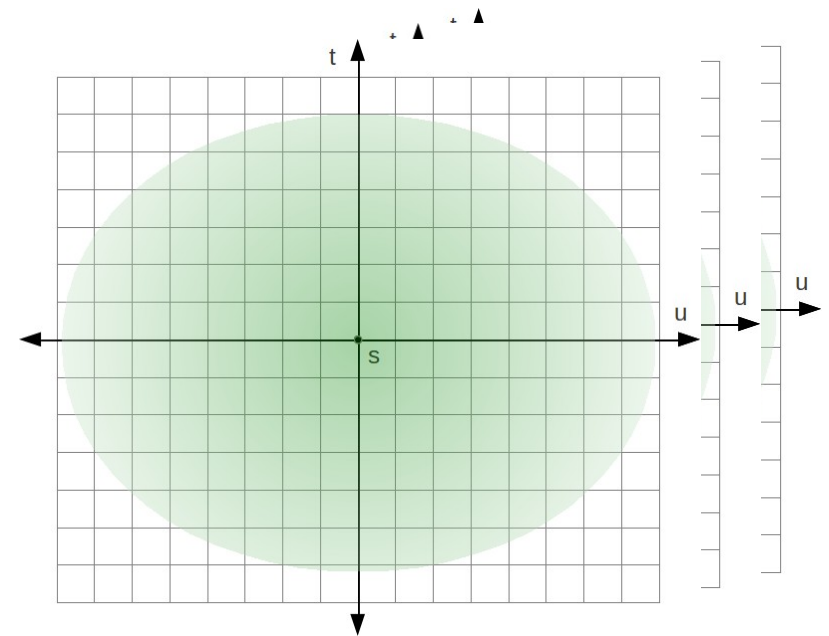
Dosisberechnung – Planewise Iteration

- Keine Synchronisation nötig
- Parallelität beschränkt durch Anzahl Planes



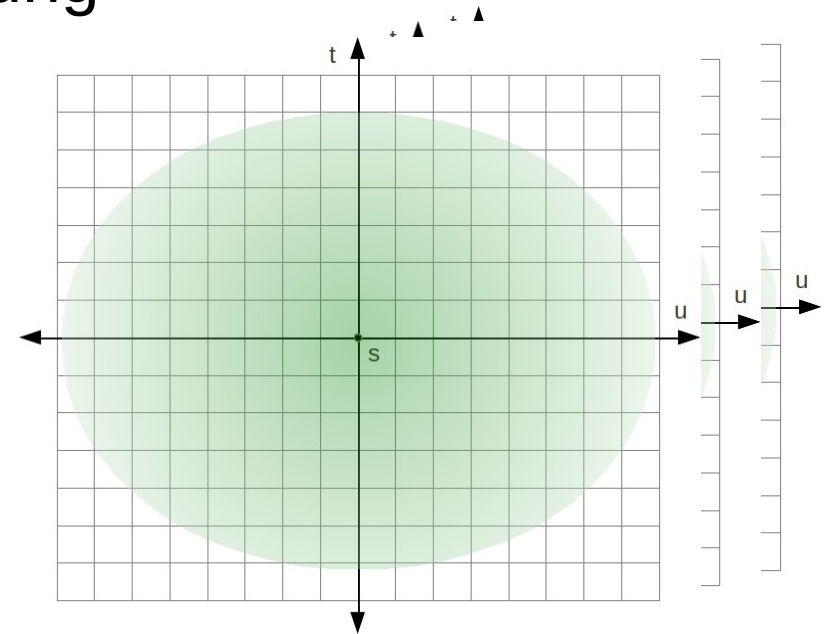
Dosisberechnung – Spotwise Iteration

- Synchronisation nötig
- Parallelität beschränkt durch Anzahl Spots



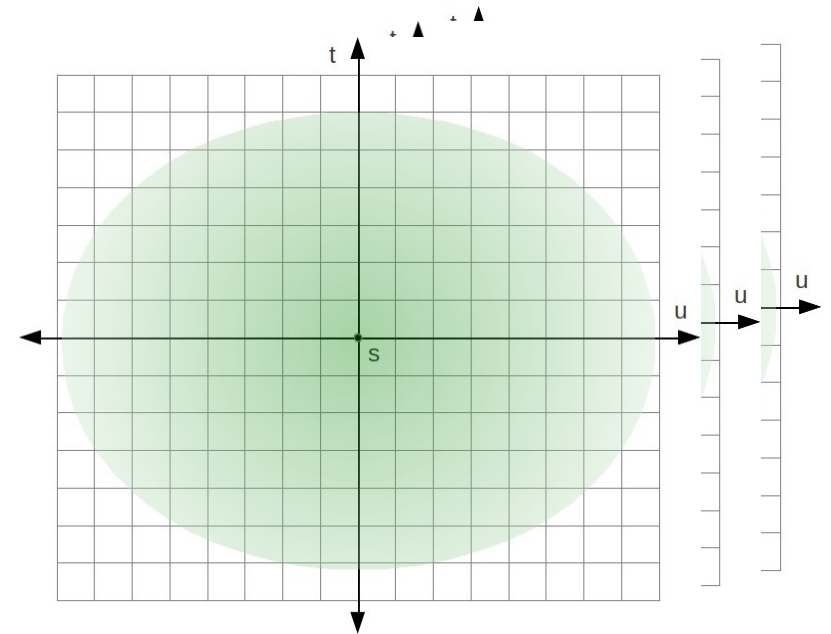
Dosisberechnung – Voxelwise Iteration

- Keine Synchronisation nötig
- Parallelität beschränkt durch Anzahl Voxels
 - Ideale Flexibilität für Arbeitsaufteilung
- Cutoff Optimierung aufwändig



Dosisberechnung – Optimierung (WIP)

- Voxelwise
- Vorberechnung der Voxel of Interests (VOI)



Dosisberechnung – Klinische Testfälle

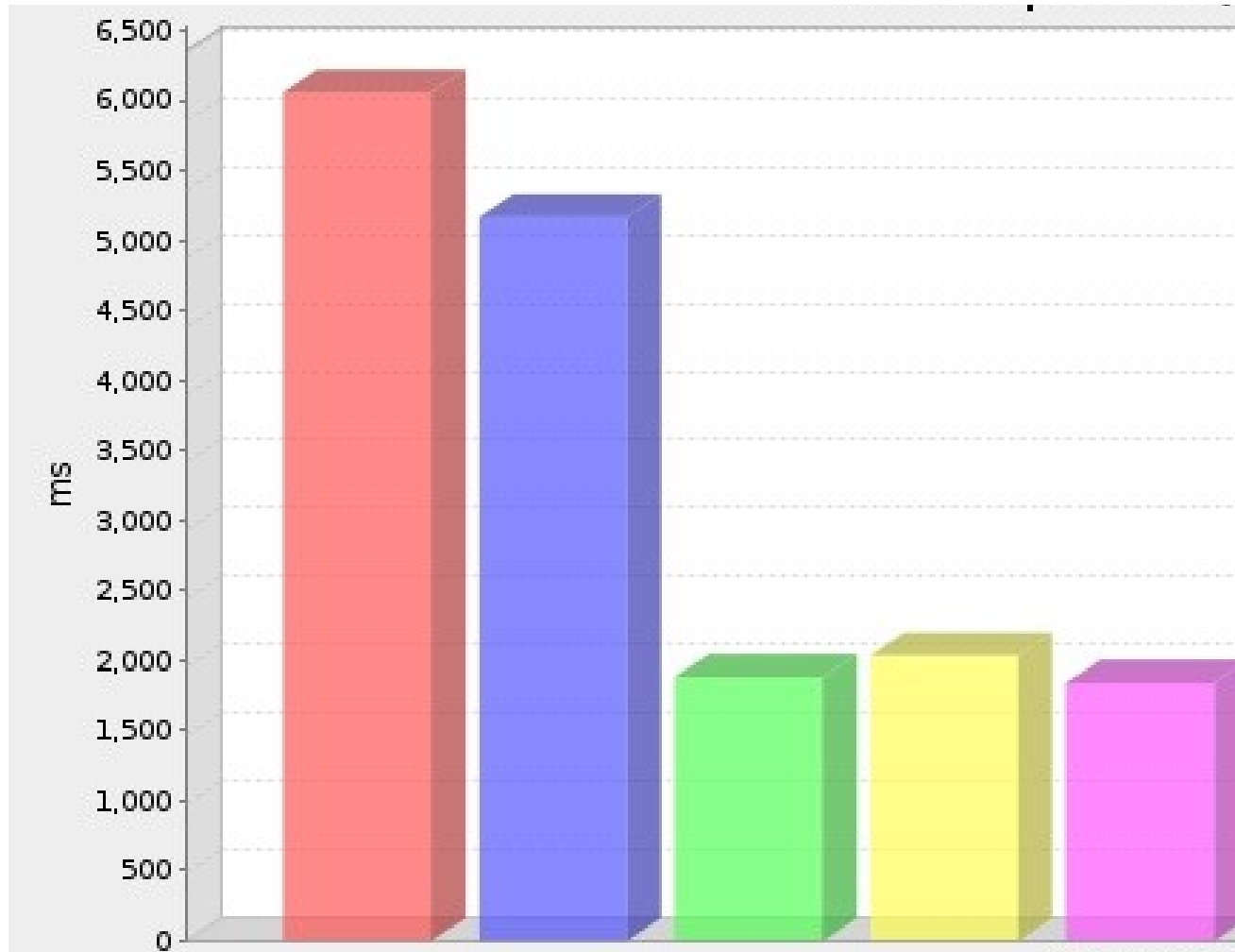
| Name | CT-Dimensions | CT-Voxels | Calculation-Grid | Calculation-Voxels | Spots |
|--------|---------------|------------|------------------|--------------------|--------|
| Small | 256,256,114 | 7'471'104 | 61,61,61 | 226'981 | 3'444 |
| Medium | 256,256,166 | 10'878'976 | 96,51,71 | 347'616 | 10'241 |
| Large | 256,256,159 | 10'420'224 | 51,51,71 | 184'671 | 51'081 |

| Name | Voxels | Affected Voxels | Avg. Spots | Total Calculations | Max Calculations | Affected Grid |
|--------|---------|-----------------|------------|--------------------|------------------|---------------|
| Small | 226'981 | 105'043 | 6.377597 | 669'921 | 781'722'564 | 46.28% |
| Medium | 347'616 | 265'791 | 10.3802 | 2'758'963 | 3'559'935'456 | 76.46% |
| Large | 184'671 | 175'288 | 5.289427 | 927'173 | 9'433'179'351 | 94.92% |

Dosisberechnung – Varianten

- „Legacy“ Java
 - Scala
 - Sequentiell
 - Planewise / Spotwise
 - OpenCL
 - nativ
 - Aparapi
-

Dosisberechnung – Ergebnisse



| driver | resultAritMean | resultAritMeanStddev | resultGeomMean | resultGeomMeanStddev | resultHarmMean | resultHarmMeanStddev |
|--|----------------|----------------------|----------------|----------------------|----------------|----------------------|
| Java Implementation | 6213.028 | 43.033 | 6213.028 | 43.033 | 6213.028 | 43.033 |
| Scala Sequential Implementation | 5299.601 | 35.777 | 5299.601 | 35.777 | 5299.601 | 35.777 |
| Scala Lock Free Planewise Parallel Implementation | 1930.34 | 45.642 | 1930.34 | 45.642 | 1930.34 | 45.642 |
| Scala CAS Spotwise Parallel Implementation | 2088.018 | 36.048 | 2088.018 | 36.048 | 2088.018 | 36.048 |
| Scala Locked Spotwise Parallel Implementation | 1890.032 | 46.124 | 1890.032 | 46.124 | 1890.032 | 46.124 |

Dosisberechnung – Fast Scala

- tailrecursion
 - Vorsicht vor Objekterzeugung
 - Typealiases
 - Value Classes
 - Interface Design (provide methods for common tasks)
 - `update(x: Int,y: Int,z: Int,f: Double => Double): ch.psi.cpt.dosecalc.impl.model.Grid`
 - `add(x: Int,y: Int,z: Int,v: Double): ch.psi.cpt.dosecalc.impl.model.Grid`
-

Dosisberechnung – Work Item

- Zweidimensionale Work Items
 - Work Item
 - Berechnet die Dosis eines Spots in einer Plane
 - Benötigt atomics
 - Andere Möglichkeiten
 - Spotwise (atomics)
 - Planewise (lock-free)
-

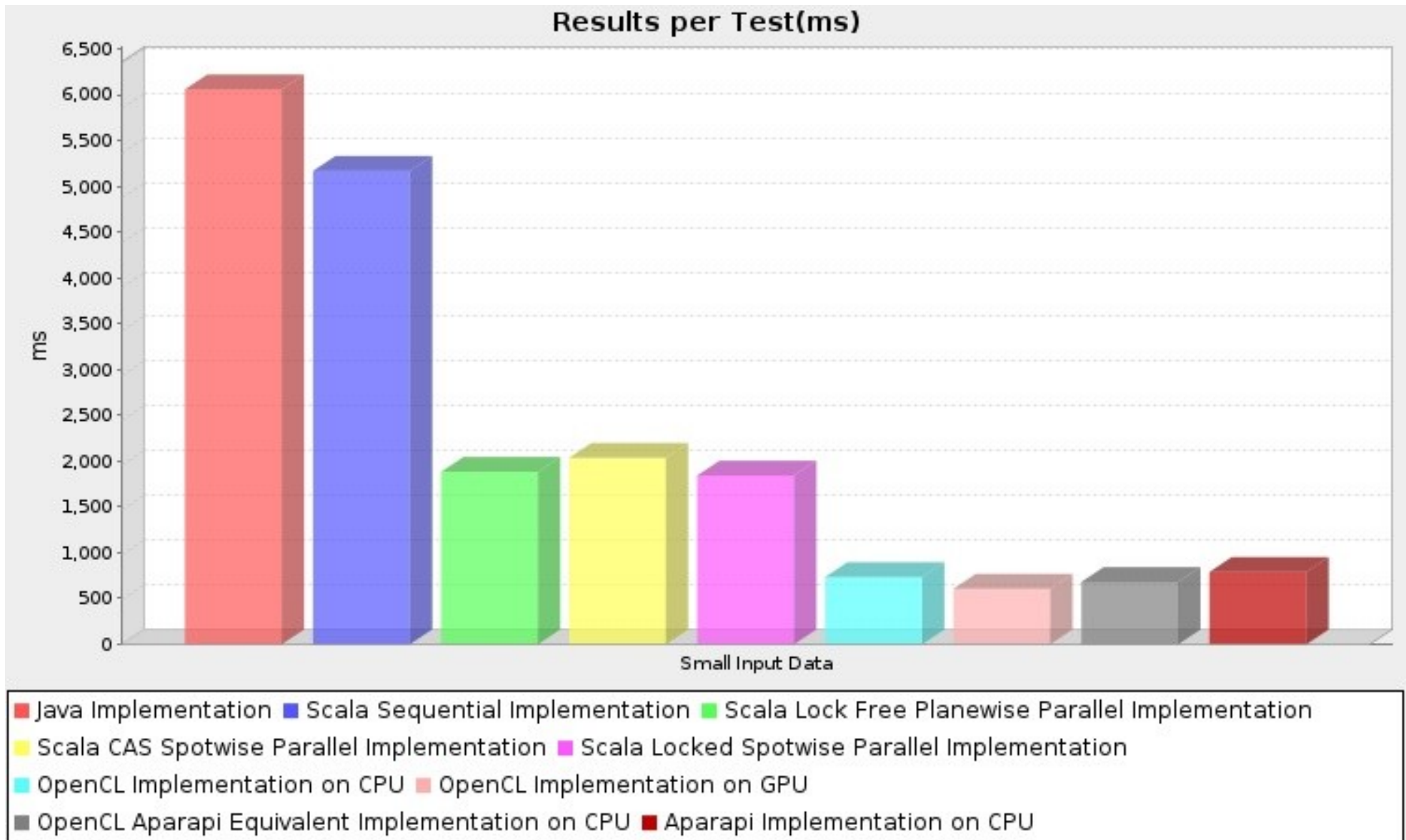
Dosisberechnung – OpenCL Memory

- Dosis / Spots im globalen Memory
- Lookup Tables in konstanten Speicher
- Keine Verwendung von lokalem Speicher

Dosisberechnung – OpenCL C Testing

- Black Box Unit Tests
 - Für die Dosisberechnung
- Testen des Modells
 - Hilfskernel welche eine einzige Funktion ausführen

Dosisberechnung – Ergebnisse



| driver | resultAritMean | resultAritMeanStddev | resultGeomMean | resultGeomMeanStddev | resultHarmMean | resultHarmMeanStddev |
|---|----------------|----------------------|----------------|----------------------|----------------|----------------------|
| Java Implementation | 6213.028 | 43.033 | 6213.028 | 43.033 | 6213.028 | 43.033 |
| Scala Sequential Implementation | 5299.601 | 35.777 | 5299.601 | 35.777 | 5299.601 | 35.777 |
| Scala Lock Free Planewise Parallel Implementation | 1930.34 | 45.642 | 1930.34 | 45.642 | 1930.34 | 45.642 |
| Scala CAS Spotwise Parallel Implementation | 2088.018 | 36.048 | 2088.018 | 36.048 | 2088.018 | 36.048 |
| Scala Locked Spotwise Parallel Implementation | 1890.032 | 46.124 | 1890.032 | 46.124 | 1890.032 | 46.124 |
| OpenCL Implementation on CPU | 754.026 | 7.584 | 754.026 | 7.584 | 754.026 | 7.584 |
| OpenCL Implementation on GPU | 621.703 | 3.977 | 621.703 | 3.977 | 621.703 | 3.977 |
| OpenCL Aparapi Equivalent Implementation on CPU | 699.152 | 8.801 | 699.152 | 8.801 | 699.152 | 8.801 |
| Aparapi Implementation on CPU | 809.903 | 77.437 | 809.903 | 77.437 | 809.903 | 77.437 |

Dosisberechnung – LOC

| Part | Java | Scala | OpenCL C | Aparapi |
|------------------|------------|------------|------------|------------|
| Dose Calculation | 104 | 107 | 104 | 165 |
| Model | 190 | 132 | 119 | 160 |
| Setup | | | 202 | 119 |
| Datenstrukturen | 46 | 45 | 30 | |
| Total | 340 | 284 | 455 | 444 |

| | Java | Scala | C |
|--------------|------------|-------------|------------|
| Total | 628 | 3601 | 499 |

Dosisberechnung – Vergleich Scala

- + JVM: Java Performance erreichbar
 - + Java Tooling: jvisualvm
 - + Abstraktion und Wiederverwendung
 - Objekterzeugungsfreudig (javap)
-

Dosisberechnung – Vergleich Aparapi

- + Gleiche Devices wie native möglich
 - + Kein C Code
 - + Auf JVM ausführbar (debugging/testing)
 - Nicht erweiterbar
 - Eingeschränkte Funktionalität (2D-Arrays)
-

Dosisberechnung – Vergleich OpenCL

- + Gesamtes OpenCL verfügbar
 - + Flexibel per preprocessor auf Device anpassen
 - + Maximale OpenCL Performance erreichbar
 - C Code
 - komplexes Testing
 - Manuelles Memory Management (JVM <-> Device)
-

Dosisberechnung – Ausblick

- Optimierung des Algorithmus auf parallele Hardware
 - Voxel of Interest (VOI) vorberechnen

Gastvorlesung

Concurrent Programming

