SIMULATING REAL-TIME FIRE FOR FIREFIGHTING TRAINING

Fire and Evacuation Modeling Technical Conference (FEMTC) 2018
Gaithersburg, Maryland, October 1-3, 2018

Christian Niemand
University of Kassel
Germany

c.niemand@uni-kassel.de

Motivation

- Why do we need a real-time fire simulation?
  - German firefighters are looking for better education and tactical training opportunities
    - Reducing attendence time at schools
    - Reducing bottlenecks for educational resources
      - e.g. Turntable ladder, Fire container/houses
    - Simulation applications for PC, VR, Web
    - Target scale is a room fire

Possible Fire Simulation Use Cases

- Search and rescue tactics
- Breathing apparatus training
- Ventilation and smoke extraction from buildings
- Tactical leadership training (group level)
- Extinguishing techniques
- Gas measurement technique/equipment

What Do We Need?

- Game/Graphics engine to visualize the simulation
- Simulation itself
  - Fast and parallel implemented algorithms
- Conversion from 3D geometry into computable data
- Hardware power: NVIDIA GeForce 780Ti or faster

Simulation/application has to run with a minimum of 30 frames/images per second!

Research Approach

- Looking for a minimal model
  - Performance is King
  - Realism is Queen
  - Every computed data costs performance
- Step by step
  - If a model for one step works in real-time...
    - ... go to the next step
    - ... add new functionality

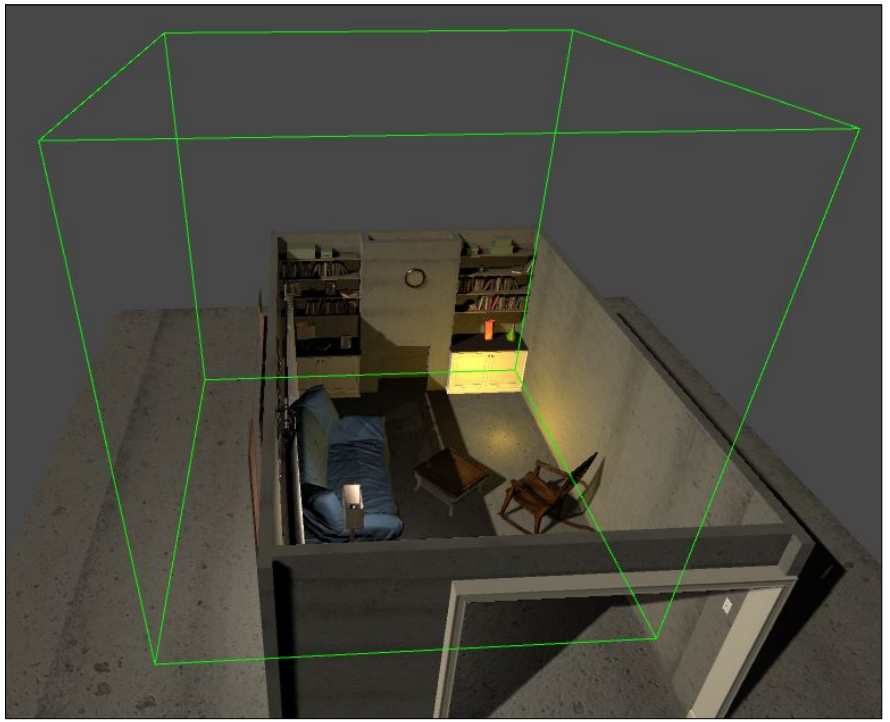
Levels Of Use Case Requirements

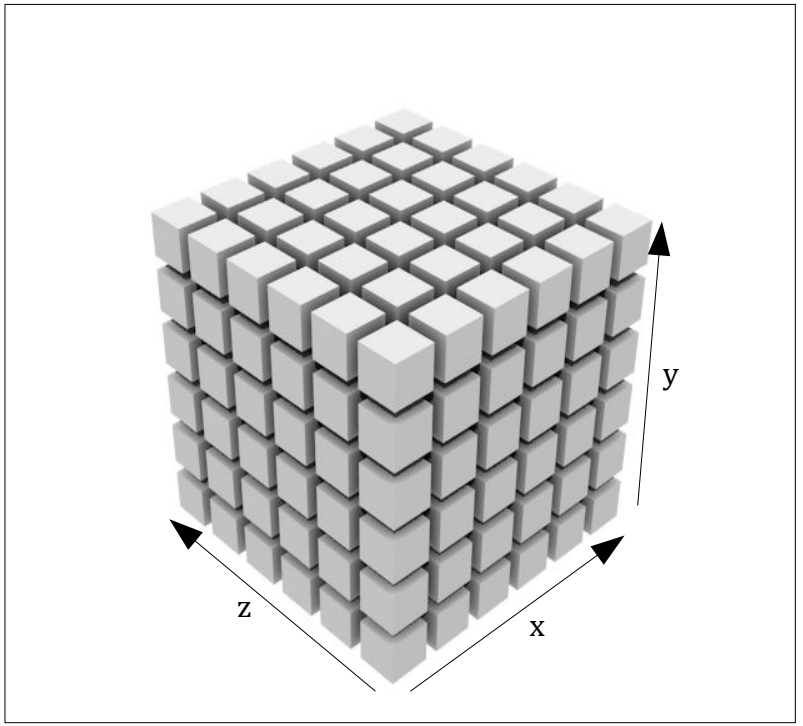
Level	Requirement	Use case
1	Visualization of smoke Smoke spreading and smoke layer Fluid dynamics Voxelization of 3D geometry Thermal imaging	Search and rescue tactics Breathing apparatus training Ventilation and smoke extraction from buildings
2	Visualization of non-spreading fire	Tactical leadership training (group level)
3	Visualization of spreading fire Visualization of fire phenomena Visualization of decomposing objects Simple fuel based pyrolysis and combustion Heat transfer Extinguishing	Extinguishing techniques (cooling down room and smoke)
4	Reactions with different fuel types Complex pyrolysis and combustion incl. Reaction speed	Measuring technique with gas measurement equipment
5	Physically based parameter settings	Simulations like FDS

Initialization (1)

Create cubic simulation volume in game engine

- Create empty data textures
- One 3D texture contains one simulation quantity (velocity, pressure, etc.)

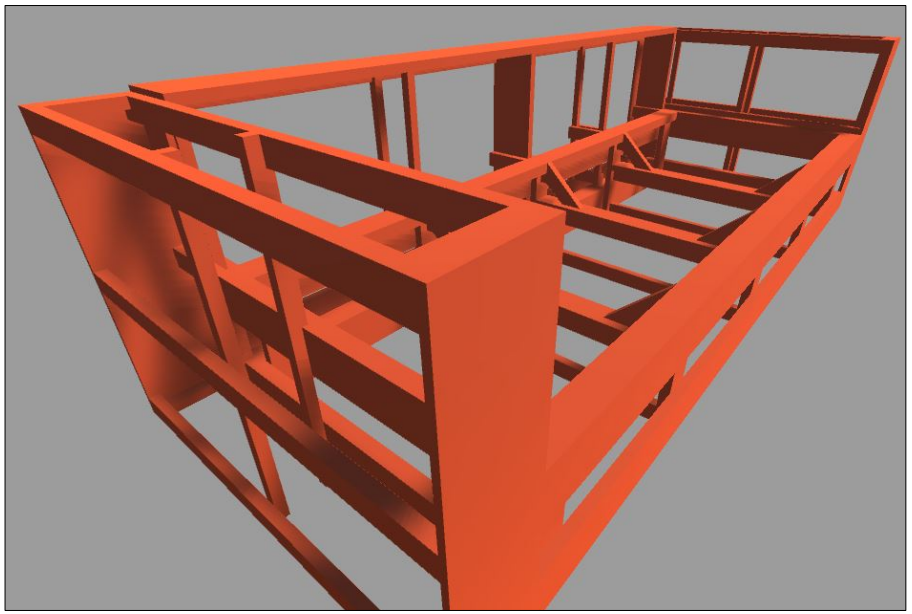


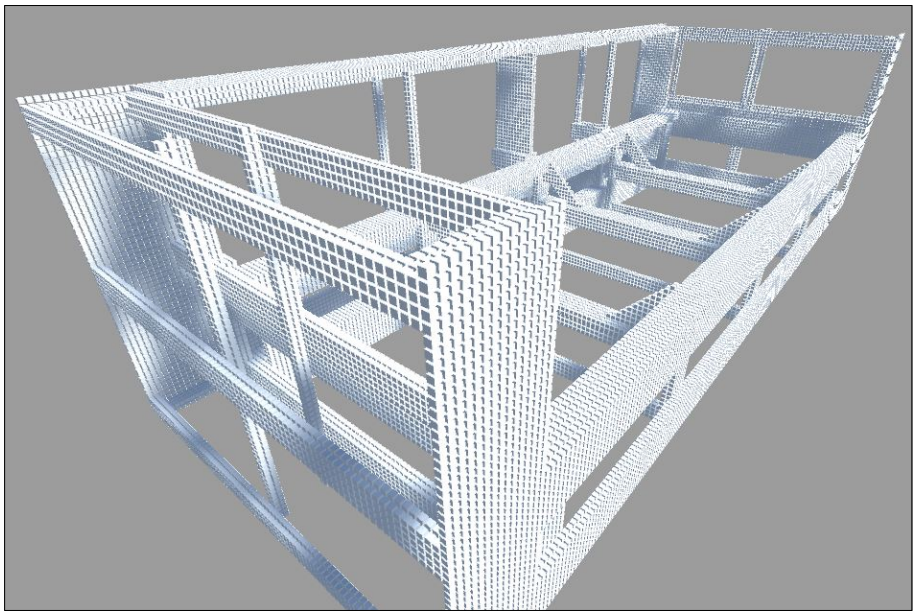


Initialization (2)

Voxelize scene geometry

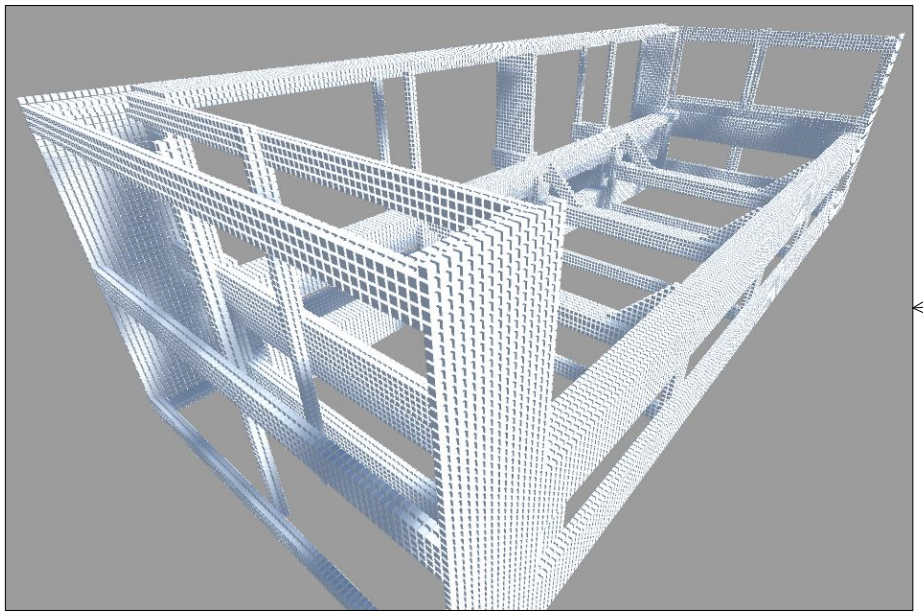
- One voxel can contain data of several quantites (temperature, material, etc.)

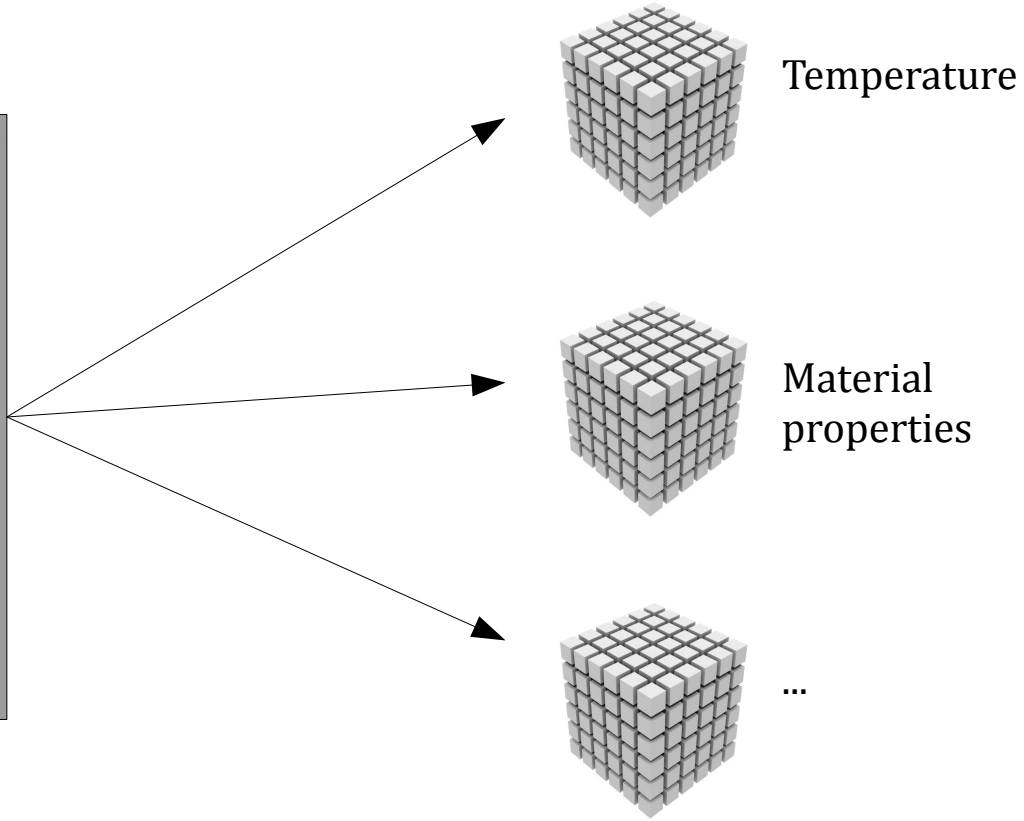




Initialization (3)

Copy voxel data into CUDA textures



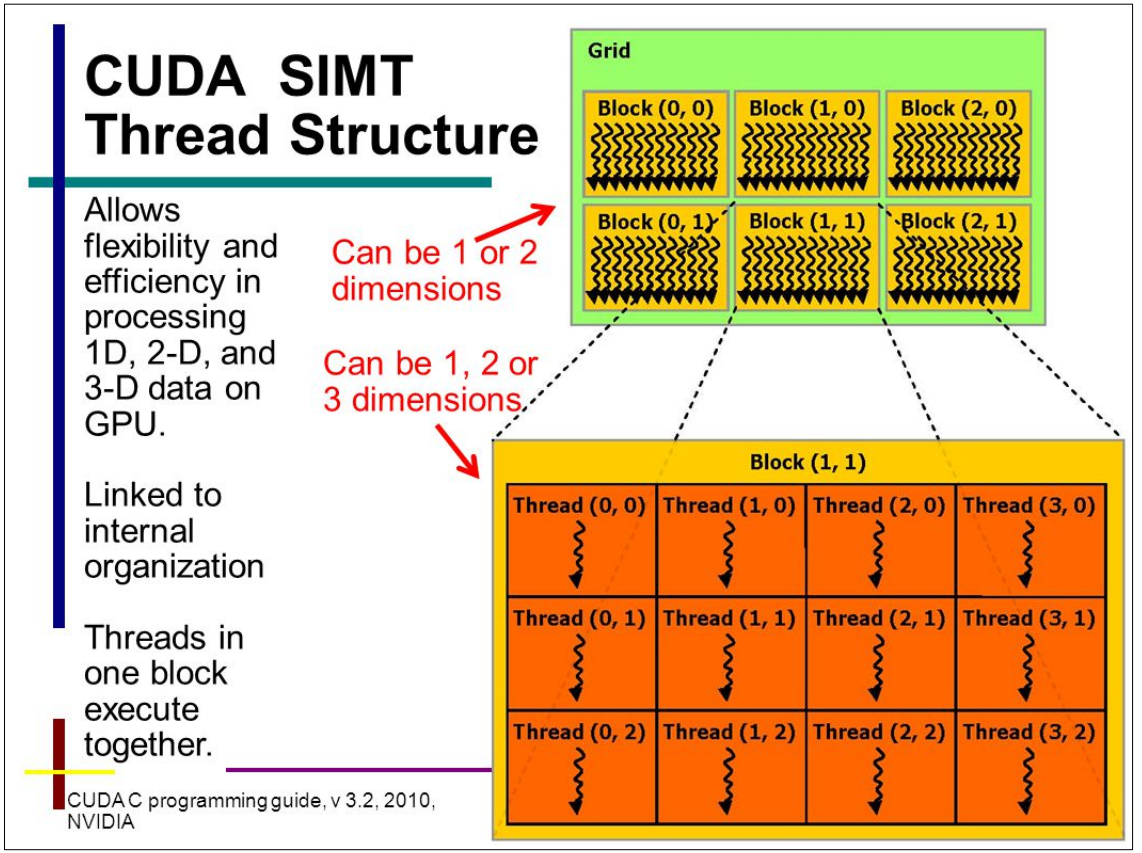


Supported Quantities Stored In Textures

Name	**Usage**
Temperature air	Contains the air temperature (all gases).
Temperature solid	Contains the temperature in solid cells.
Heat sources	Heat sources are overwriting temperature cells every frame.
Oxygen	Contains the amount of oxygen in air cells.
Fuel solid	Contains the amount of solid fuel in solid cells.
Fuel gas	Contains the amount of gaseous fuel in air cells.
Fuel type (Material)	Defines the behavior of cells.
Light	Contains the light emission produced by combustion.
Smoke	Contains the amount of smoke produced by combustion.
Velocity	Contains the velocity field which is used for transport.
Divergence	Necessary for calculating the next velocity field.
Pressure	Necessary for calculating the next velocity field.
Render Texture	Used by game engine to render all data.

Simulation Loop (1)

Parallel approach with NVIDIA CUDA



CUDA C programming guide, v3.2, 2010, NVIDIA

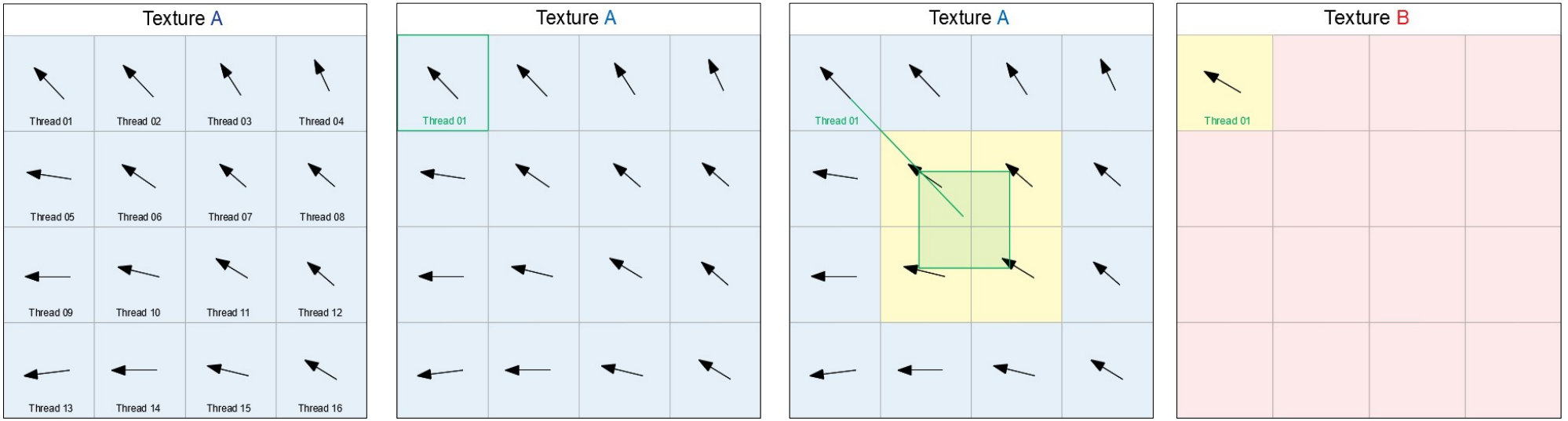
Simulation Loop (2)

```
1  while simulation is running do
2      Update game engine runtime data in CUDA plugin
3      Transport vector and scalar fields
4      Perform conduction
5      Perform heat transfer
6      Perform radiation
7      Perform pyrolysis
8      Perform combustion
9      Add buoyancy to velocity field
10     Check boundary velocity (no slip)
11     Calculate new velocity field
12     Convert simulation data into RGBA render texture
13 end
```

Transport (1)

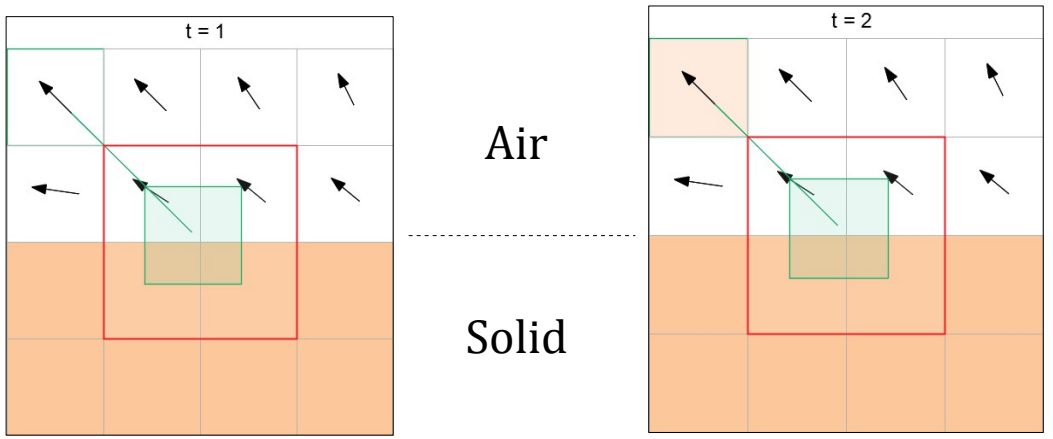
- Trace thread cell velocity back depending on dt (frame time)
- Take texture sample
- Write new value into thread cell

Based on articles of
Jos Stam and GPU Gems



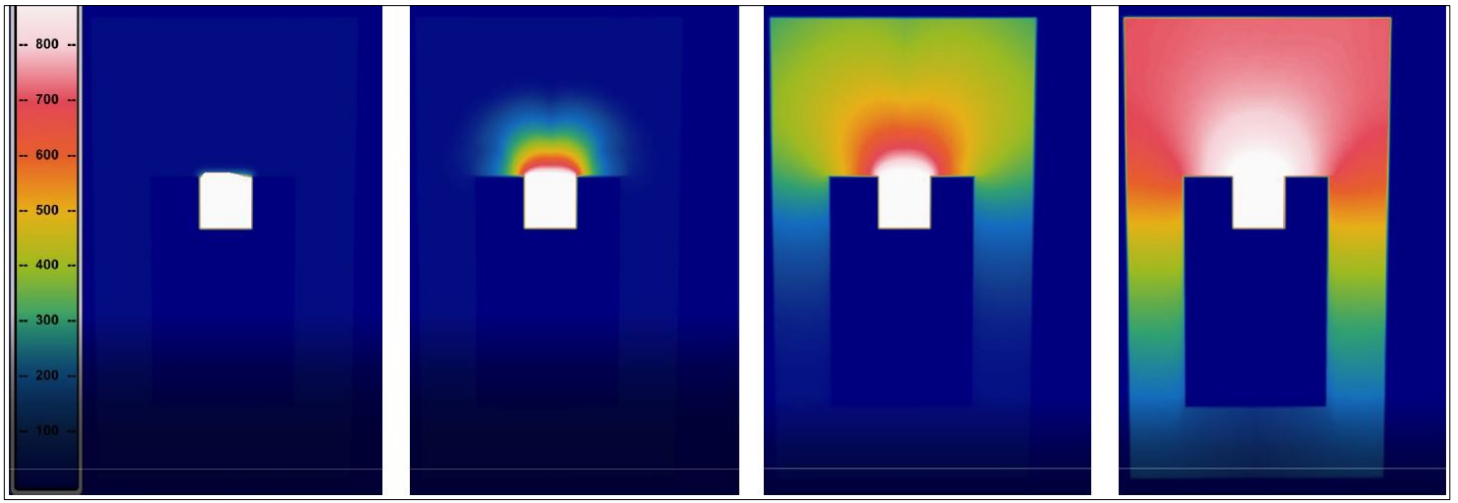
Transport (2)

- Problem: Transported quantities (e.g. temperature) can be extracted from solid objects by texture sampling
- Solution: Separate quantity into an air and solid texture



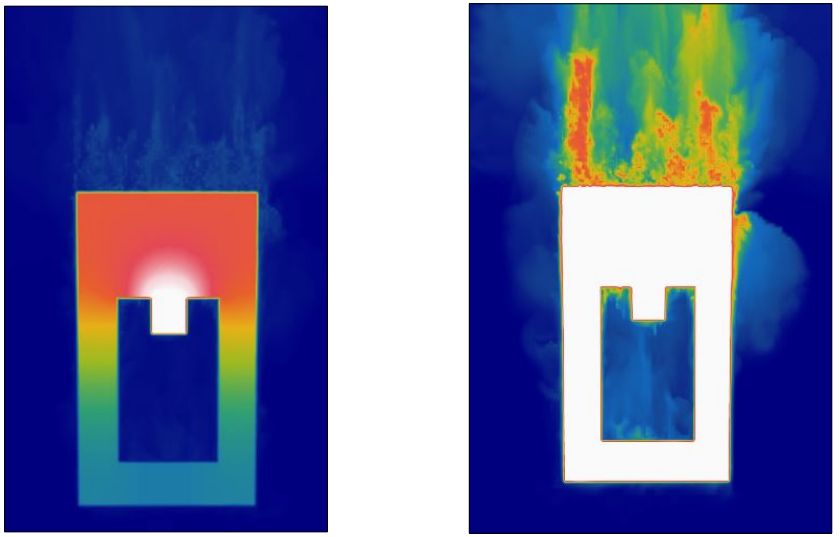
Conduction

- Based on the conductibility of temperature equation
- Describes thermal diffusivity in homogeneous and isotropic materials
  - Material attributes are the same in every voxel of one object
  - No directions in conductivity



Heat Transfer

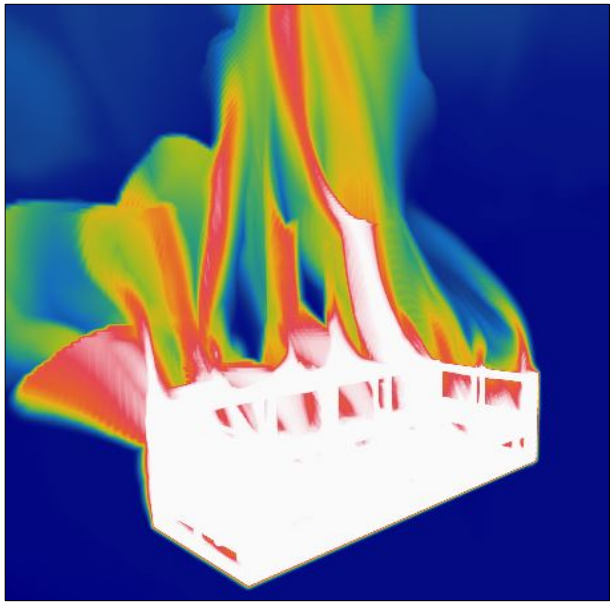
- In our simulation heat transfer describes the process when temperature is exchanged between cells at solid and gaseous borders
- Only the exchange direction from solid to gaseous cells is implemented
- Heat transfer is one process to heat up gaseous cells which results in buoyancy



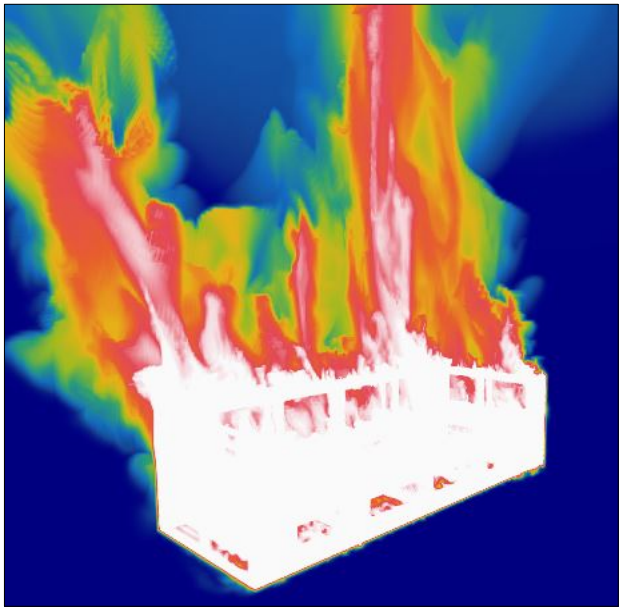
Heat transfer with different scaling for thermal imaging

Buoyancy

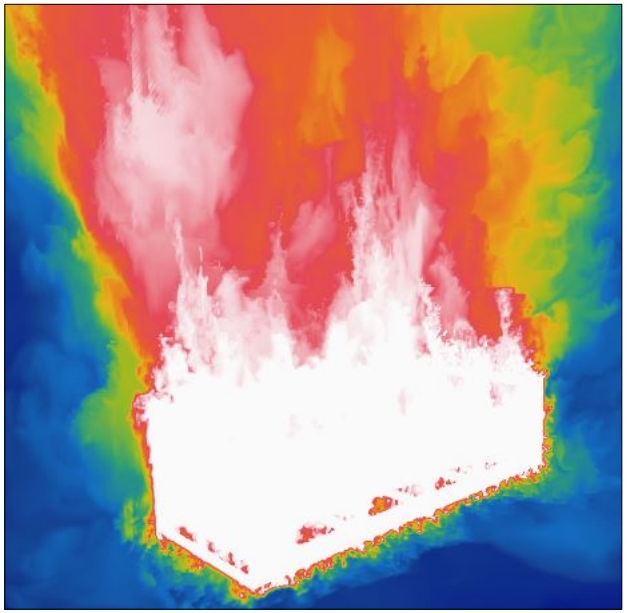
- Laminar buoyancy is based on cell temperature
- Turbulent buoyancy is base on temperature of neighbor cells
- Mixed buoyancy interpolates between both



Laminar



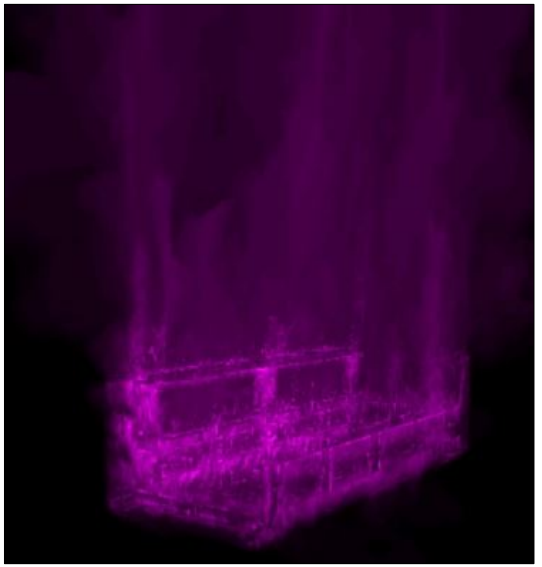
Mixed



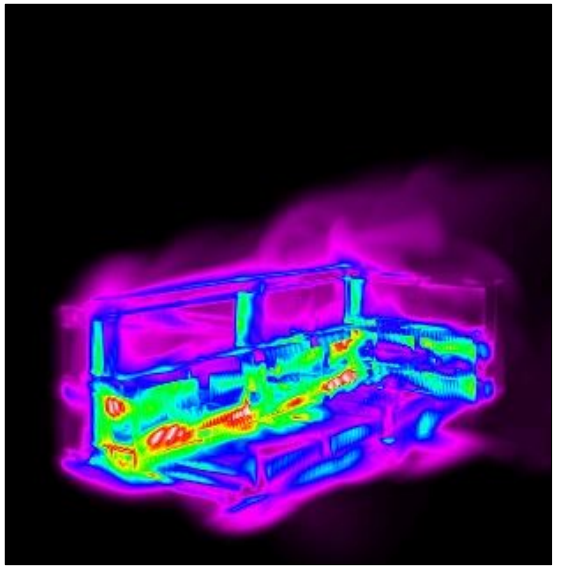
Turbulent

Pyrolysis

- Depending on the cell temperature ...
- Solid fuel is converted into gaseous fuel
  - Currenty inner cells are ignored



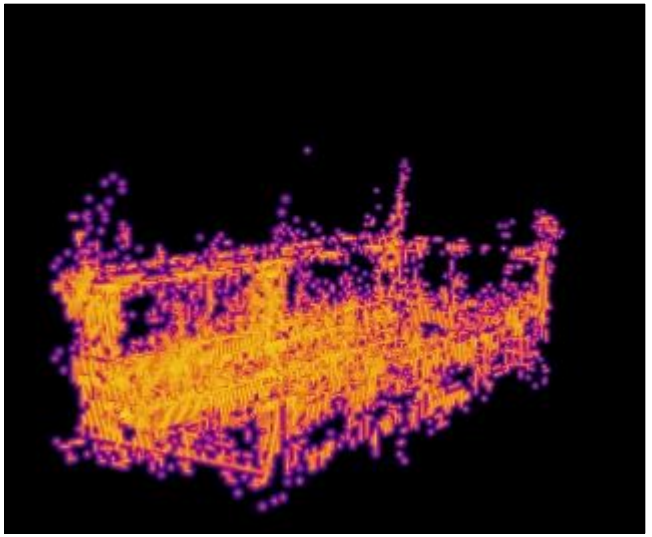
With heat transfer and buoyancy



Without heat transfer and buoyancy

Combustion

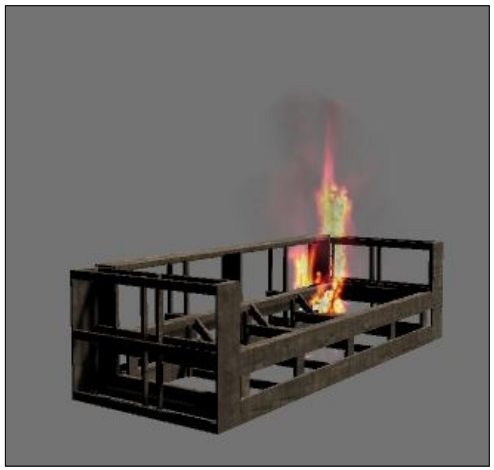
- Cell has an oxygen and gaseous fuel concentration within a reactive explosion range
- Cell temperature is above ignition temperature
- Cell is touched by a flame
- Converts oxygen and gaseous fuel into light, smoke, radiation, temperature and products



Combustion intensity

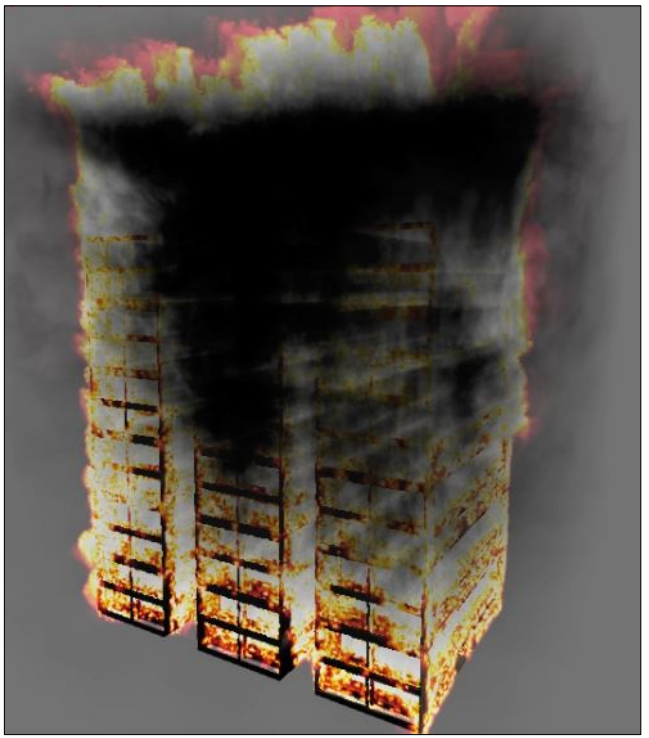
Result: Burning Couch Frame

- Couch starts ignition at heat source (back right corner)
- Temperature is spreding and starts pyrolysis
- Combustion starts where oxygen and gaseous fuel can react
- Pyrolysis ends when all solid fuel is gone




Result: Stress Test

- 18 stacked couch frames and 4 heat sources



Runtime performance

Simulations using a 256^3 grid running with:

- 35 frames per second on a NVIDIA TitanXp
  - Intel i9 2.9Ghz
  - 64 GB RAM
- 32 frames per seconds on a NVIDIA GeForce 780Ti
  - Intel i7 3.6 Ghz
  - 16 GB RAM
- **TitanXp expectations were much higher! (50fps)**
  - **Bottlenecks!**

Future Plans

- Heat transfer from gas to solid
- Radiation (existing model to real-time)
- Decomposing objects (existing model to real-time)
- Extinguishing process
- Reproducing fire phenomena (rollover, flashover)
- Reproducing couch burning experiment (full extend)
- Reproducing room fire
- Using several GPUs to simulate one scenario

Conclusion (1)

We found a model which allows a simple real-time fire simulation

- Transport, Conduction, Heat transfer solid to gas, Pyrolysis, Combustion, Buoyancy
- 10 Quantities
- Resolution: 256^3
- Hardware: NVIDIA GeForce 780Ti or better

Conclusion (2)

Pending work will cost performance but ...

- Till now we did not spent much time on optimization
  - Find bottlenecks!
- Simulation volume can be reduced (e.g. 128^3)
- Using multiple GPUs to simulate one scenario by connecting multiple simulation volumes

Questions?

c.niemand@uni-kassel.de