Simulation Ideen-Verbreitung

Projektvorstellung

Arne Struck, Jonathan Werner, Manuel Börries

Universität Hamburg, Fachschaft Informatik, Praktikum paralleles Programmieren

24. September 2014

(Grobe) Simulation von Entwicklung konkurrierender Ideen in einer begrenzten Welt.

Idee

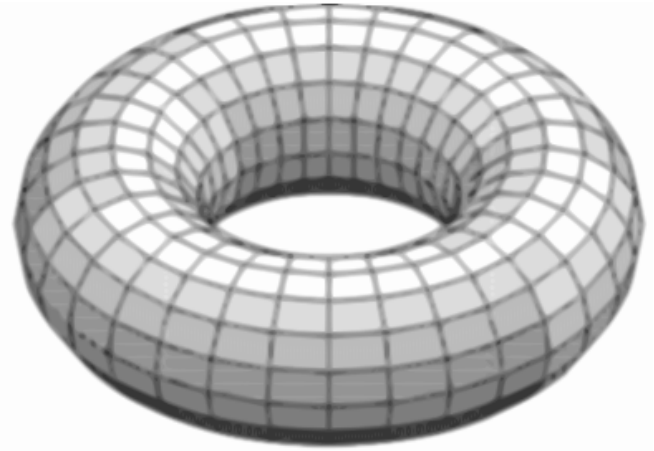
- Qualität
- Komplexität
- Weltanschauung

Mensch

- Idee
- Weltanschauung

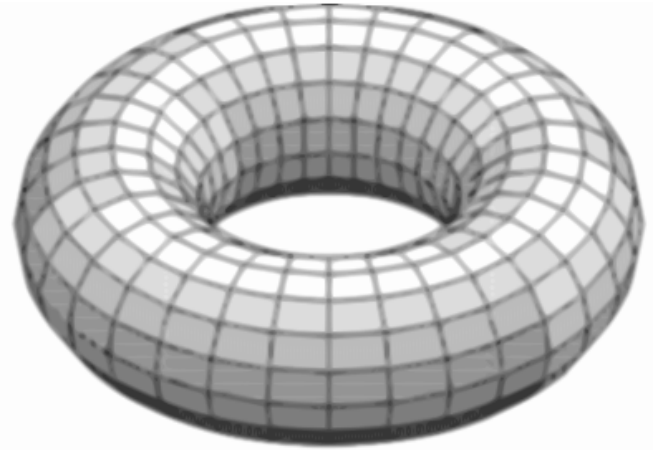
Welt & Bewegung

Die Welt

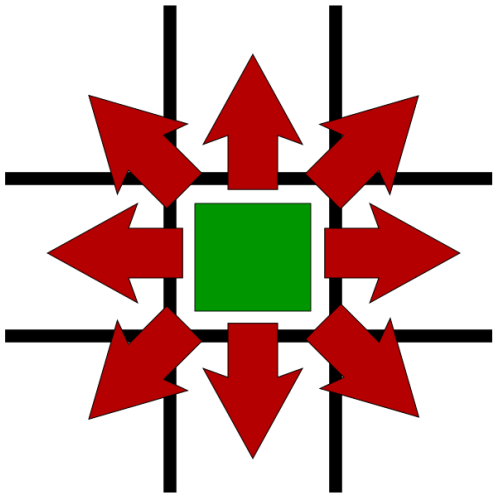


Welt & Bewegung

Die Welt

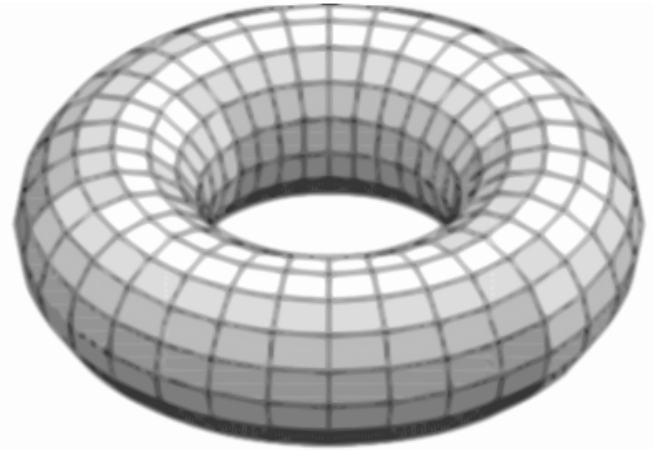


Bewegungsziele

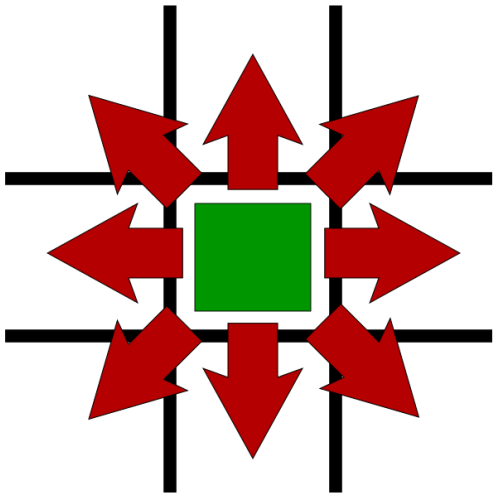


Welt & Bewegung

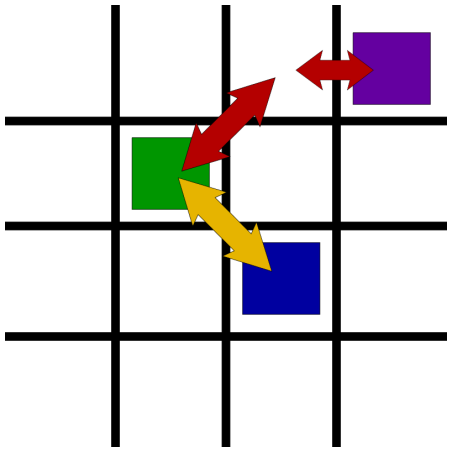
Die Welt



Bewegungsziele



Kommunikation



Kommunikation

3-Phasen:

1. Kompatibilitätscheck
2. Evaluation des Gewinners
3. Aufstellung der neuen Merkmale des Verlierers

Mutation

Qualität

- Wahl der Mutationsrichtung
- Mutation der Qualität
- Kaskadierend der Komplexität

Mutation

Qualität

- Wahl der Mutationsrichtung
- Mutation der Qualität
- Kaskadierend der Komplexität

Weltanschauung

- Wahl der Mutationsrichtung
- Mutation des Idee-Wertes
- Mutation des Mensch-Wertes
- Differenzcheck

Ablauf

- Initialisierung des Feldes
- Zufälliger Spawn der Menschen mit mehrheitlich geringen Qualitätswerten
- Beginn der Simulationsschleife für n Schritte
  - Mutationsevaluation
  - Kommunikationsversuch
  - Bewegung
- Ende der Schleife

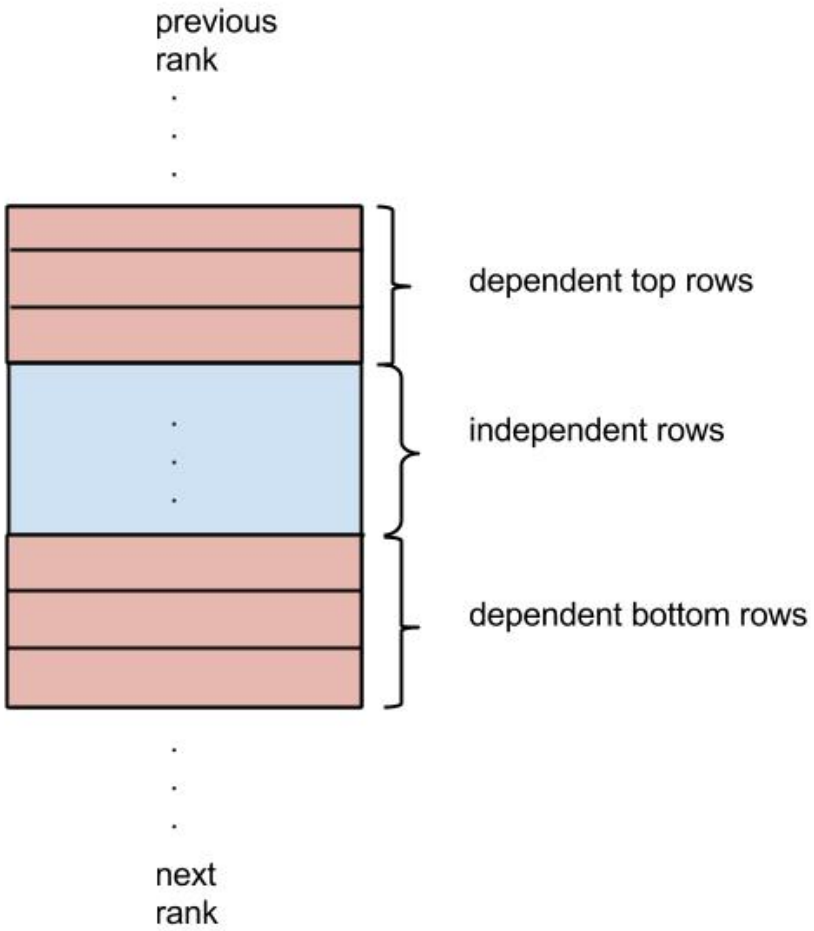
Ideen

placeholder

Kommunikation & Gewinner Berechnung

Parallelisierungsschema

Parallelisierungsschema



1. Bewegung der independent ideas
2. Bewegung der top dependent ideas, Kommunikation dieser
3. Bewegung der bottom dependent ideas, Kommunikation dieser

Parallelisierungsschema

```
#define send_ideas(ideas_arr, to, tag, req)
  MPI_Isend(ideas_arr, num_cols,
    mpi_idea_type, to, tag, MPI_COMM_WORLD, &req)

#define receive_ideas_into(ideas_arr, from, tag, req)
  MPI_Irecv(ideas_arr, num_cols,
    mpi_idea_type, from, tag, MPI_COMM_WORLD, &req)
```

Parallelisierungsschema

```
#define mpi_define_idea_type()
   int blocklengths[5] = {1,1,1,1,1};
   MPI_Datatype types[5] = {MPI_INT,MPI_INT,MPI_INT,
       MPI_INT, MPI_INT};
   MPI_Datatype mpi_idea_type;
   MPI_Aint offsets[5];
   offsets[0] = offsetof(Idea, a);
   offsets[1] = offsetof(Idea, b);
   offsets[2] = offsetof(Idea, c);
   offsets[3] = offsetof(Idea, h);
   offsets[4] = offsetof(Idea, empty);
     MPI_Type_create_struct(5, blocklengths, offsets
         , types, &mpi_idea_type);
     MPI_Type_commit(&mpi_idea_type);
```

Profiling

HIER EIN BILD VON MESSUNGEN

Tracing

idk, ob wir das machen wollen

Ablauf

HIER BITTE EIN GUTES ABLAUF-GIF PLS

Ergebnisse

- Qualität nimmt über die Zeit zu
- Obwohl andere selten vollständig entfernt bilden 2-3 Ideen eine Majorität aus
- Qualität/Elaboriertheit nimmt über die Zeit zu
- Es bleiben einige Menschen mit Ideen niedriger Qualität
- Selten: Durch Mutation entwickelt sich eine verdrängte Idee zur dominanten