The Art of Iterating: Update-Strategies in ABS

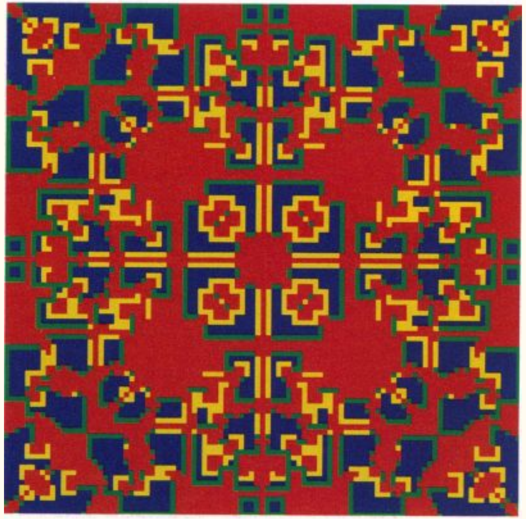
Supervisors: Dr. Peer-Olaf Siebers, Dr. Thorsten Altenkirch

Jonathan Thaler

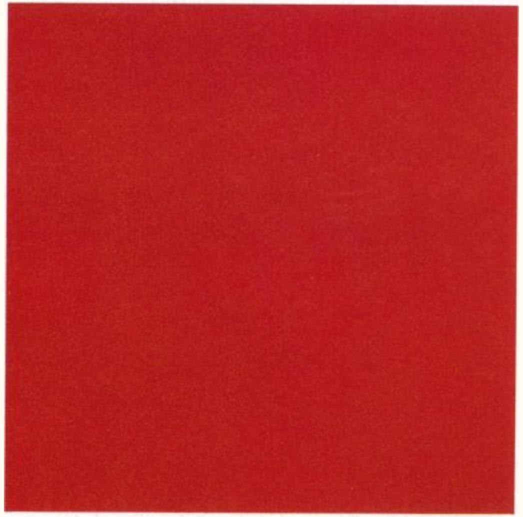
Motivation

Prisoner-Dilemma on 2D-Grid[1]

- Cooperate or Defect
- 1 Defector @ center
- Neighbourhood plays
- Highest payoff wins



synchronous[2]



asynchronous[2]

[1] Nowak, M. A., and May, R. M. Evolutionary games and spatial chaos. Nature 359, 6398 (Oct. 1992), 826–829.
[2] Huberman, B. A., and Glance, N. S. Evolutionary games and computer simulations. Proceedings of the National Academy of Sciences 90, 16 (Aug. 1993), 7716–7718.

Message & Aim

Select the update-strategy which reflects the semantics of the model.

- Present new terminology
- Compare three very different languages

New Terminology

Properties of ABS

1. Iteration-Order

   Sequential or **Parallel**?

2. Global Synchronization

   Yes or **No**?

3. Thread of Execution

   Separate or **Shared**?

4. Message-Handling

   Immediately or **Queued**?

5. Visibility of Changes

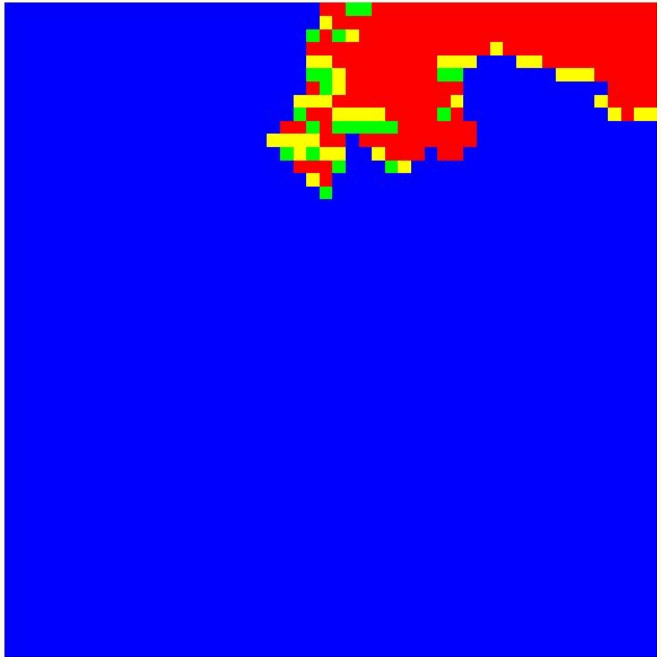
   In-Iteration or **Post-Iteration**?

6. Repeatability

   Deterministic or **Non-Deterministic**?

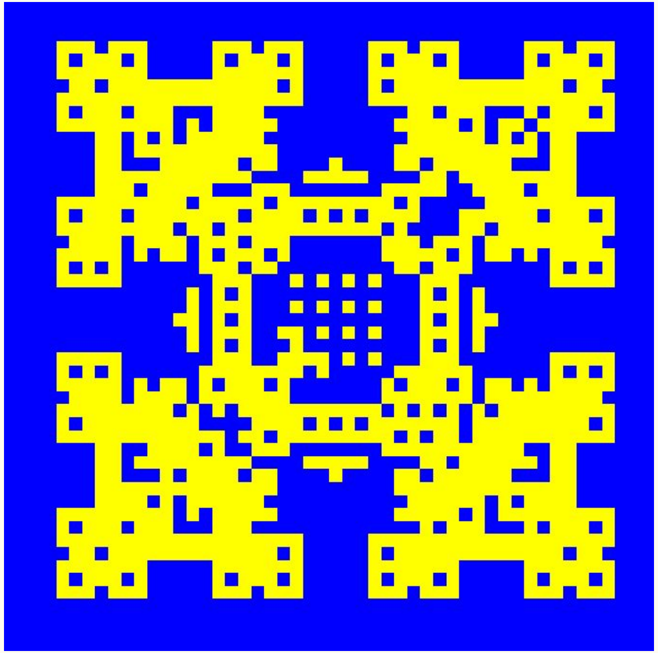
I Sequential Strategy

- Global synchronization
- Update Agents sequentially
- Changes visible immediately
- Shared global thread



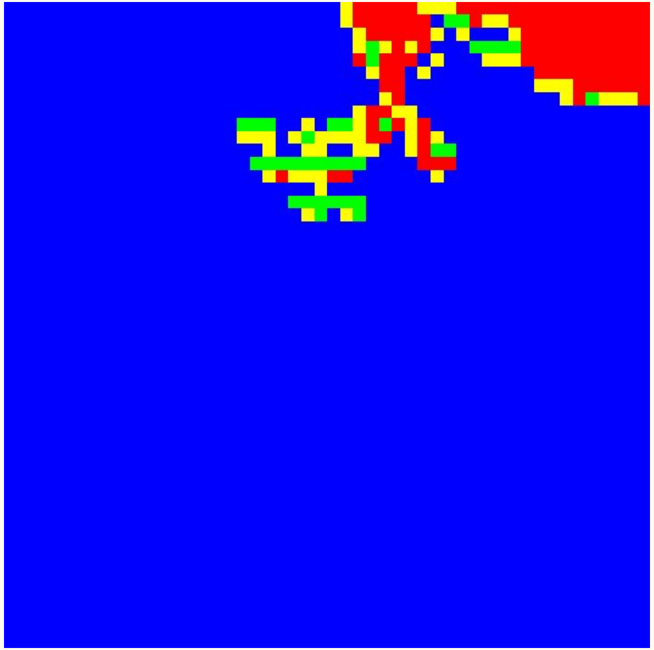
II Parallel Strategy

- Global synchronization
- Update Agents parallel
- Changes visible NEXT iteration
- Shared global / separate thread



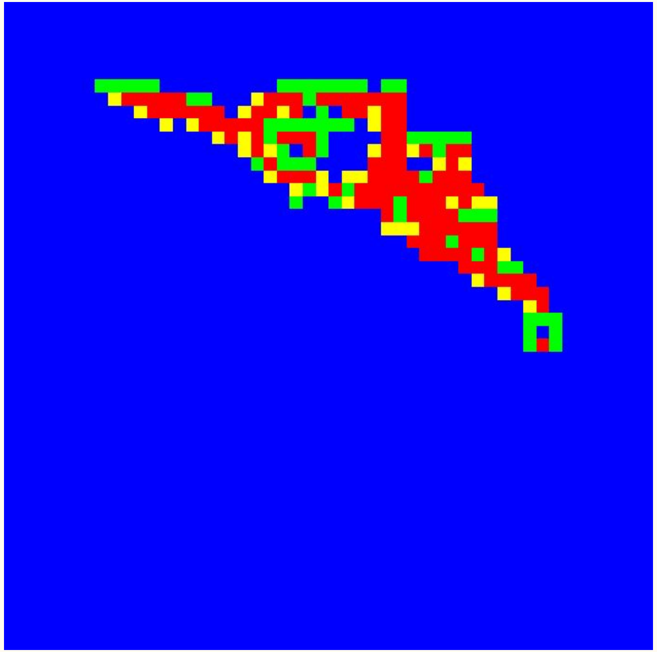
III Concurrent Strategy

- Global synchronization
- Update Agents parallel
- Changes visible immediately
- Separate thread



IV Actors Strategy

- **No** global synchronization
- Update Agents parallel
- Changes visible immediately
- Separate thread



Language Comparison

Java

- **Ease-Of-Use:** all Strategies faithfully
- **Benefits:** widespread, high-performance
- **Deficits:** parallelism and concurrency guidance
- **Natural:** Sequential Strategy

Haskell

- **Ease-Of-Use:** all Strategies faithfully

- **Benefits:** static type-system, parallelism and concurrency

- **Deficits:** immediate message-handling, performance

- **Natural:** Parallel & Concurrency Strategy

Scala with Actors

- **Ease-Of-Use:** only implemented Actor Strategy
- **Benefits:** elegant solutions, concurrency
- **Deficits:** Nondeterministic
- **Natural:** Actor Strategy

Conclusion

Conclusion

- Properties of ABS & Update-Strategies
- Haskell surprised!
- Actor-Model promising in ABS
- **Update-Strategy must match Model**

Q & A