High-Dimensional Battleship- comparsion of shooting strategies

Christopher Schnick
Dennis Kreittner
Dirk Pflüger

University of Stuttgart
Germany

Institute for Parallel and Distributed Systems (IPVS)
Simulation Software Engineering
Universitätsstr. 38, D-70569 Stuttgart, Germany
schniccr@ipvs.uni-stuttgart.de
kreittds@ipvs.uni-stuttgart.de

Motivation and Approach

Objective: **Investigate** the **functionality** of high-dimensional battleships. **Realize** and **compare** possible shooting **strategies** based on discretization methods. The general idea for this topic comes from [1].

- Adapt classic battleship to a **high-dimensional hypercube**.
- Redefine terms like **ship, fleet** and shooting **strategies**.
- Evaluation of strategies requires many sample fleets and even more ships.
- A **Monte-Carlo approach** reduces the computation cost significantly while still allowing a decent impression of the quality of the strategy.
- Firstly we use an algorithm to compute the number of shots v to sink a random ship.
- With this algorithm we can then calculate the **distribution function** $P(\mathbf{T}^{\Omega_L}_{\text{strat}} \leq v)$ of the strategy for a sample Ω_L.
- The Monte-Carlo approach now assumes that this distribution resembles the distribution for all possible ships.

$$P(\mathbf{T}^{\Omega_L}_{\text{strat}} \leq v) \approx P(\mathbf{T}^{L_{all}}_{\text{strat}} \leq v) \tag{1}$$

- Note: Careful when discretizing ships as shown in [1], there are many continuous ships that represent the same discrete ship as seen in figure 1.

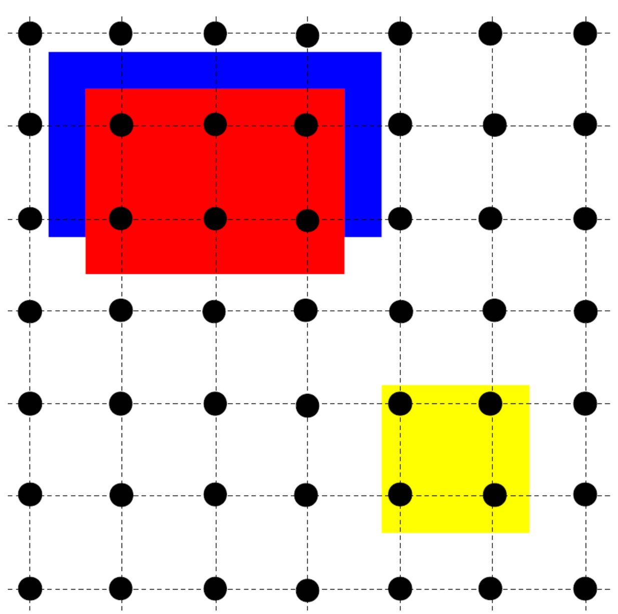

Figure 1: Difference continuous to discrete ships

- The red and blue continuous ships in figure 1 assign to the same discrete ship represented by the six dots.

Strategies

For shooting strategies we want to orientate us at **discretization methods** since they are easily adapted to high dimensions. For that matter we came up with **five different strategies**. In general grid startegies use the following partitioning:

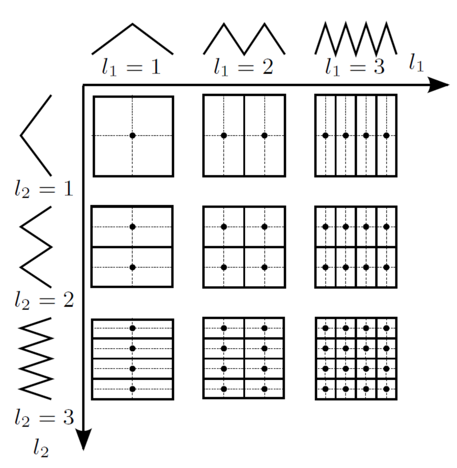

Figure 2: Halving of the grid coordinates

- **Random Strategy**: Shoots at a random cell of the hypercube. This is the closest strategy to the reality since most battleship players target random cells.
- **Full Grid Strategy**: Shoots cells according to the discretization given by a full grid.

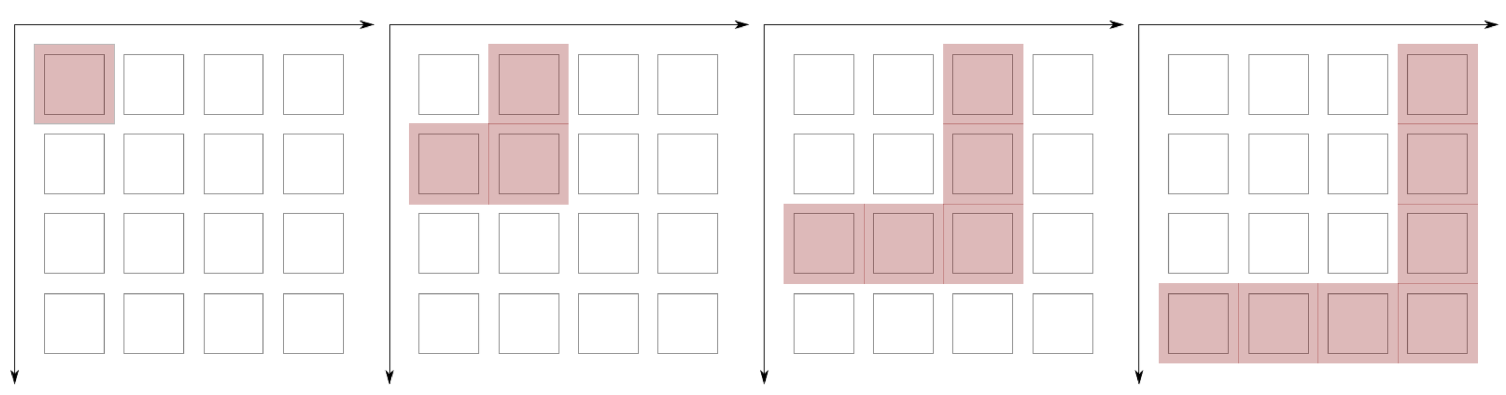

Figure 3: Structure of Full Grid Strategy

- **Sparse Grid Strategy**: Shoots cells according to the discretization given by a sparse grid.

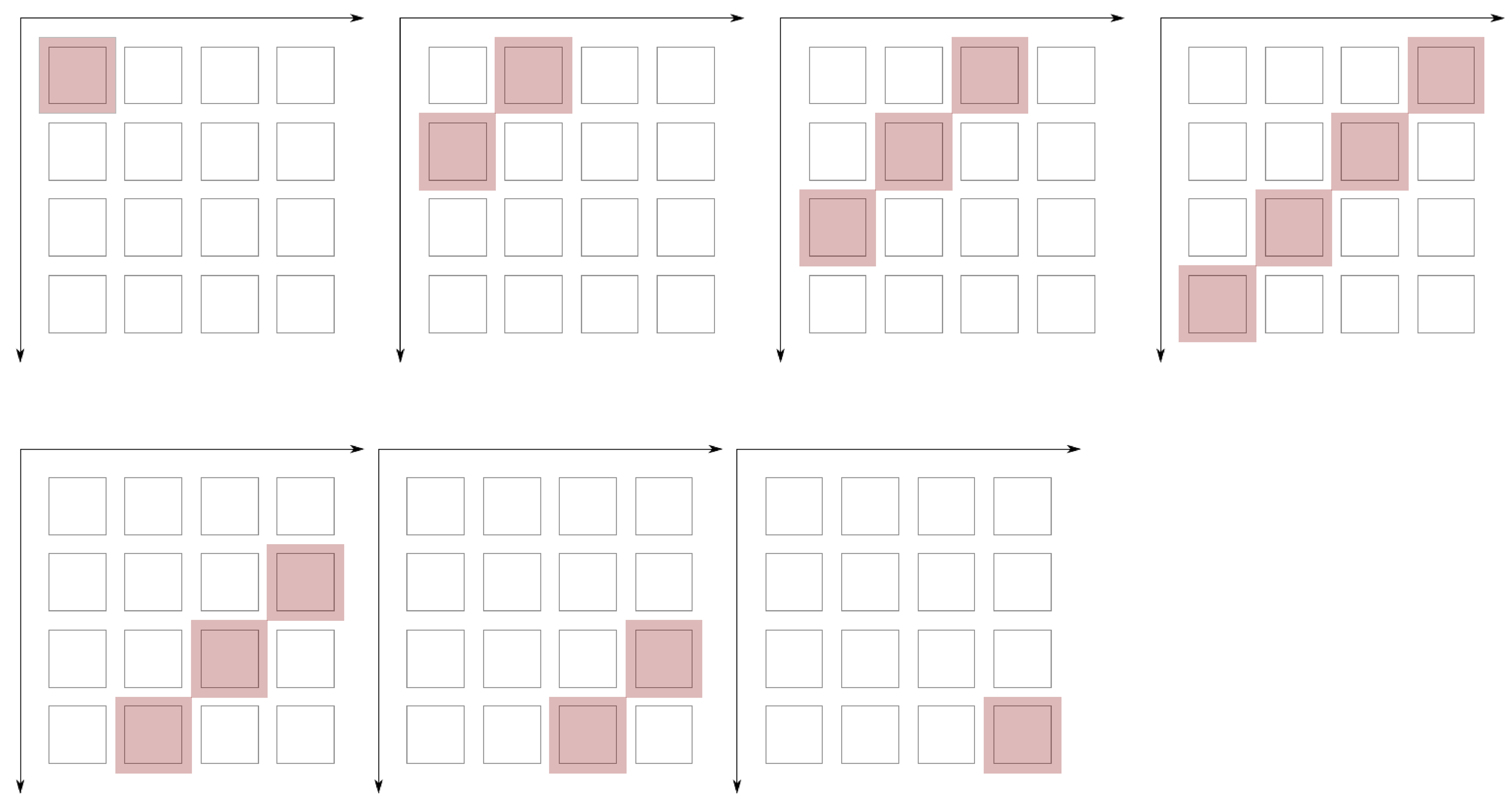

Figure 4: Structure of Sparse Grid Strategy

- **Sobol Strategy**: This strategy shoots according to the pseudorandom Sobol number sequence which is based on polynomials in $\mathbb{Z}_2$. The definitions and algorithms are based on [2].
- **Halton Strategy**: Here the strategy is based on onedimensional number sequences of fractions of primes which we combine to multidimensional coodinates.

Results

After computing the results for the different strategies with **varying dimensions** and **sample sizes** we can compare them now.

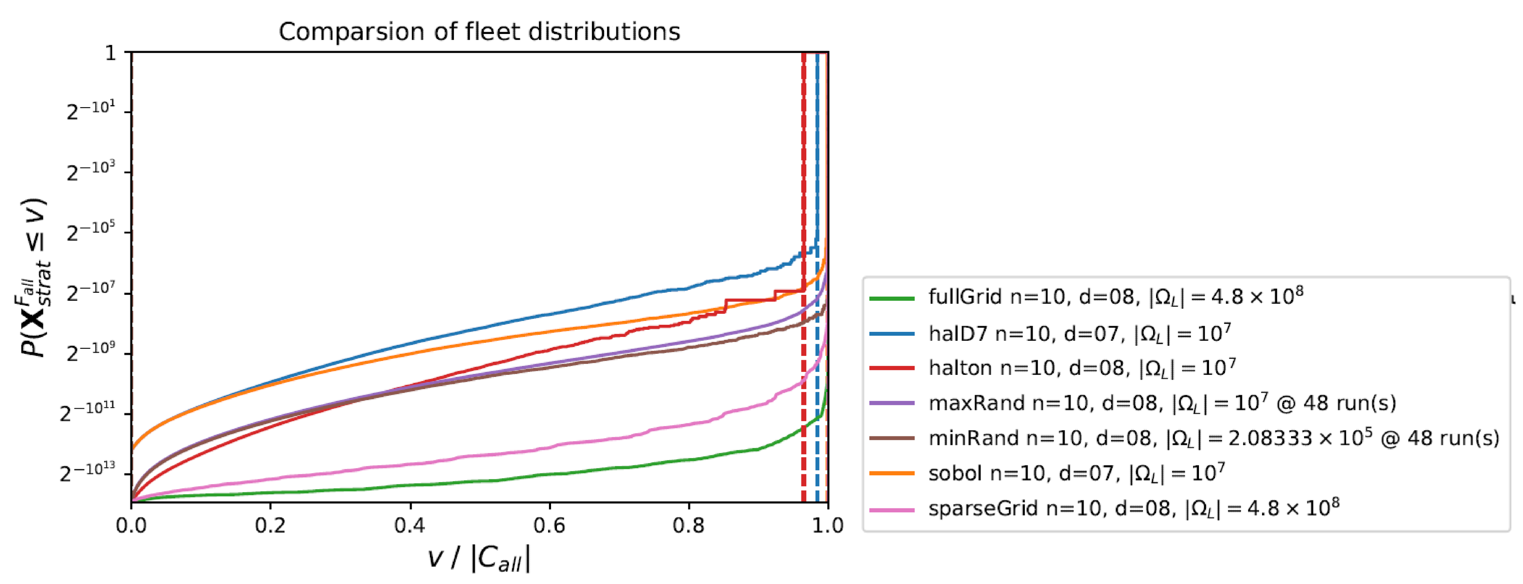

Figure 5: Comparsion Strategies for random fleet

In figure 5 we see, that the Halton Strategy for 7 dimensions overtakes Sobol. This implies the same for 8 dimensions. Which is why we conclude that the **Halton Strategy performs the best** out of all the tested strategies. Note: The goal of the curves is to reach the top as fast as possible. This conclusion is also confirmed by the expected value of shots that a strategy needs in which Halton generally needs $3 \cdot 10^6$ **less shots** in 8 dimensions than other strategies.

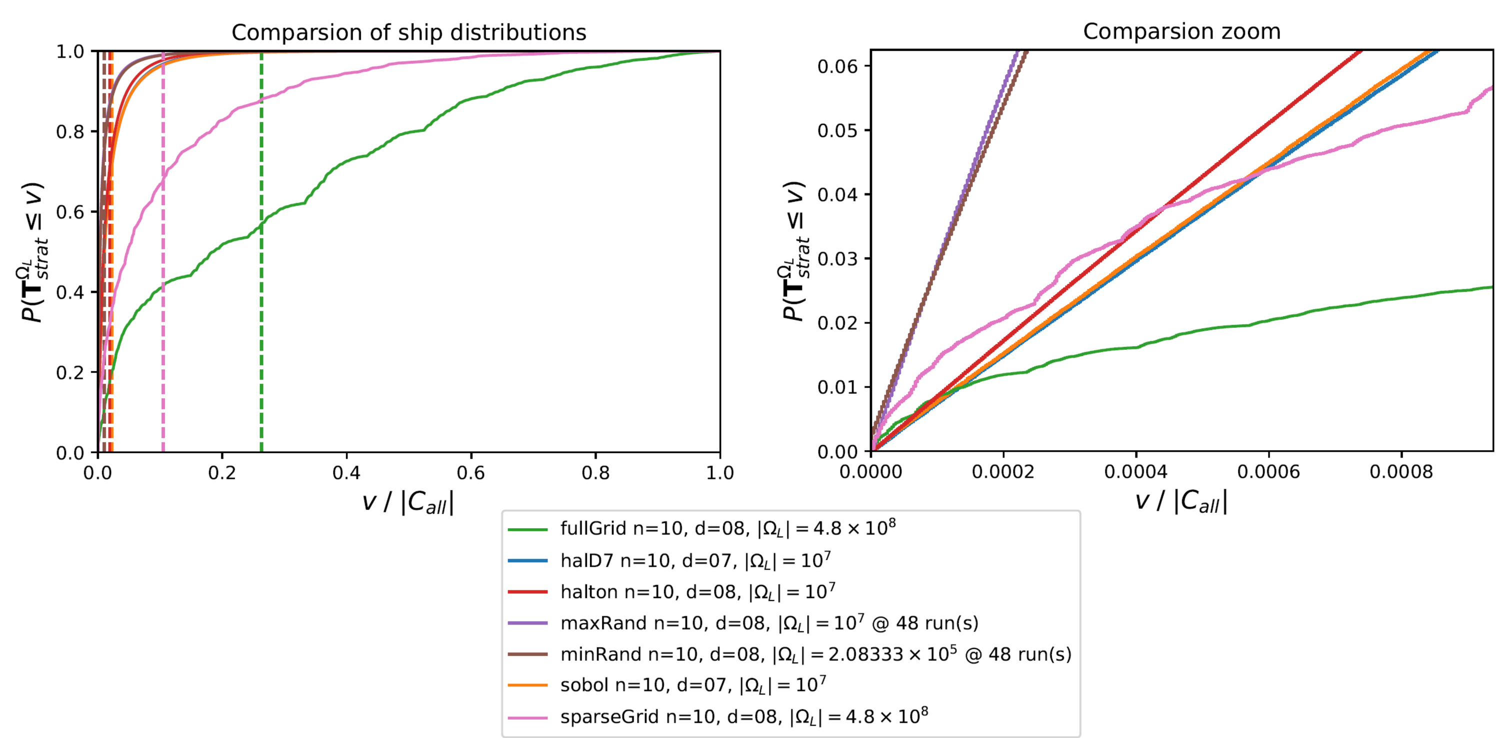

Figure 6: Comparsion Strategies for random ship distribution

References

[1] Mehlbeer F.
Hierarchische methoden am beispiel von schiffe versenken, 06 2013.

[2] Joe S. and Kuo F. Y.
Remark on algorithm 659: Implementing sobol's quasirandom sequence generator.
ACM Transactions on Mathematical Software, 29:49–57, 2003.