W4. Perception & Situation Awareness & Decision making

- Robot Perception for Dynamic environments: Outline & DP-Grids concept
- Dynamic Probabilistic Grids – Bayesian Occupancy Filter concept
- **Dynamic Probabilistic Grids – Implementation approaches**
- Object level Perception functions (SLAM + DATMO)
- Detection and Tracking of Mobile Objects – Problem & Approaches
- Detection and Tracking of Mobile Objects – Model & Grid based approaches
- Embedded Bayesian Perception & Short-term collision risk (DP-Grid level)
- Situation Awareness – Problem statement & Motion / Prediction Models
- Situation Awareness – Collision Risk Assessment & Decision (Object level)

DP-Grids – *How to compute P(OV | Z C) in practice?*

Theoretical solving *(see previous session)*

$$P(O\,V\,|\,Z\,C) = \lambda \sum_{A\,O^{-1}\,V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the possible antecedents A
and their states (O^{-1} V^{-1})

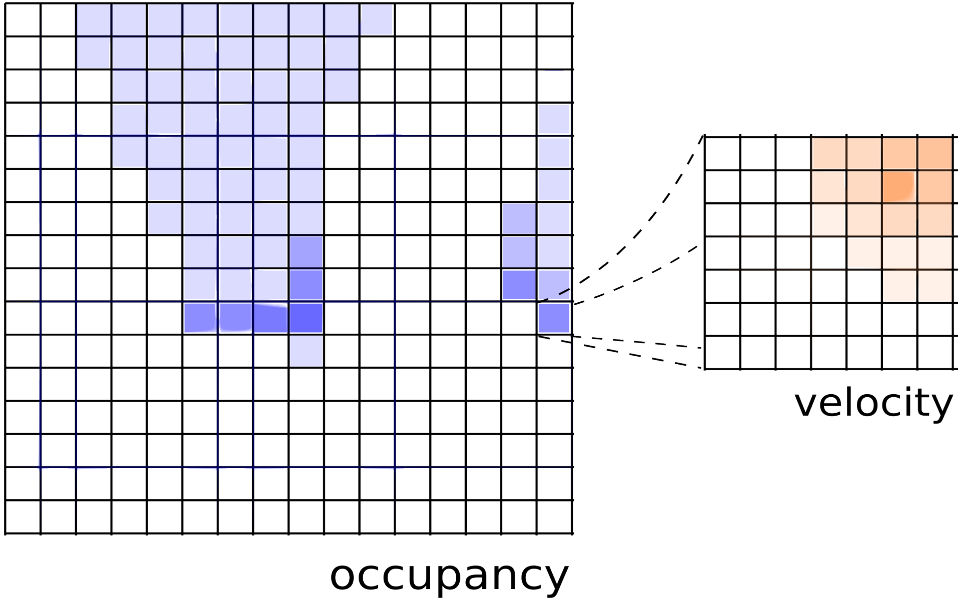
With: $P(C\ A\ O\ O^{-1}\ V\ V^{-1}\ Z) = P(A)\ P(O^{-1}\ V^{-1}\ |\ A)\ P(O\ V\ |\ O^{-1}\ V^{-1})$

$$P(C\ |\ A\ V)\ P(Z\ |\ O\ C)$$

Computing this expression is difficult in practice
=> Huge range of possible antecedents
=> Strongly depends on grid size & velocity range

DP-Grids – *How to compute P(OV | Z C) in practice?*
Initial approach: The classic BOF

- Regular grid
- Transition histograms for every cell (for representing velocities)

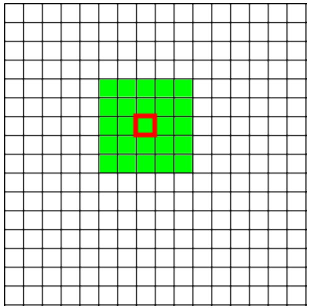


DP-Grids – *How to compute P(OV | Z C) in practice?*

Initial approach: The classic BOF

$$P(O\,V\,|\,Z\,C)=\lambda\sum_{A\,O^{-1}\,V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the ~~possible~~ antecedents A
and their ~~states~~ (O^{-1} V^{-1})



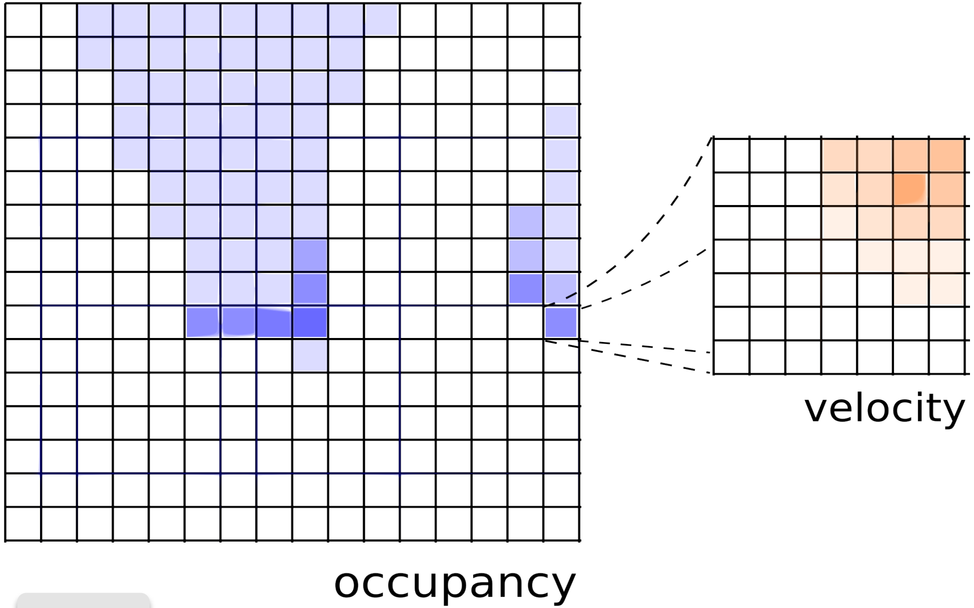
Practical computation:

→ *Sum over the **neighborhood,** with a **single possible velocity per antecedent A** of equation:*

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\,P(O^{-1}\,V^{-1}\,|\,A)\,P(O\,V\,|\,O^{-1}\,V^{-1})\,P(C\,|\,A\,V)\,P(Z\,|\,O\,C)$$

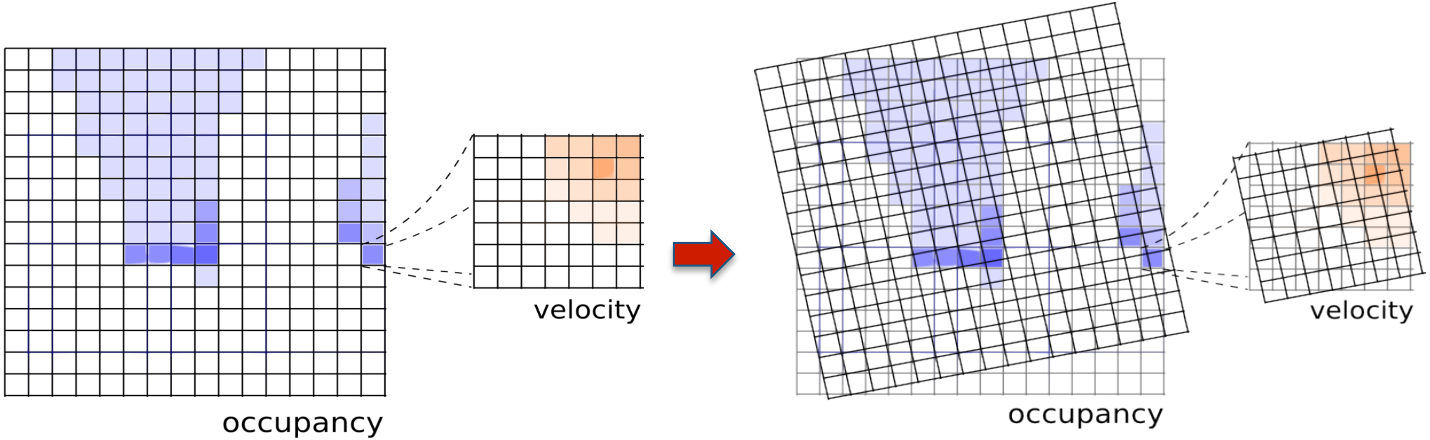
DP-Grids – *How to compute P(OV | Z C) in practice?*
Initial approach: Drawbacks

- ***Velocity histogram*** *needs to be accurate on both* ***low*** *&* ***high velocities***
  - ➔ *Its resolution has to be high, while being* ***mostly empty***
  - ➔ *It requires a* ***large memory size*** *(but in practice the* ***accuracy is weak****)*



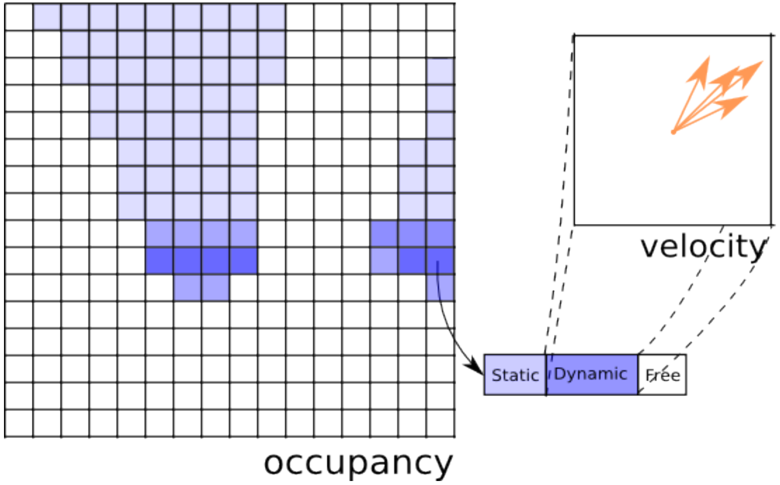
DP-Grids – *How to compute P(OV | Z C) in practice?*
Initial approach: Drawbacks

- **Temporal aliasing** ➔ *Due to **update / real frequencies synchronization***
- **Spatial aliasing** (moving grid) ➔ *High complexity due to 4-dimension interpolation, approximations required in practice*



DP-Grids – *How to compute P(OV | Z C) in practice?*
Improvement: Hybrid Sampling BOF (HSBOF)

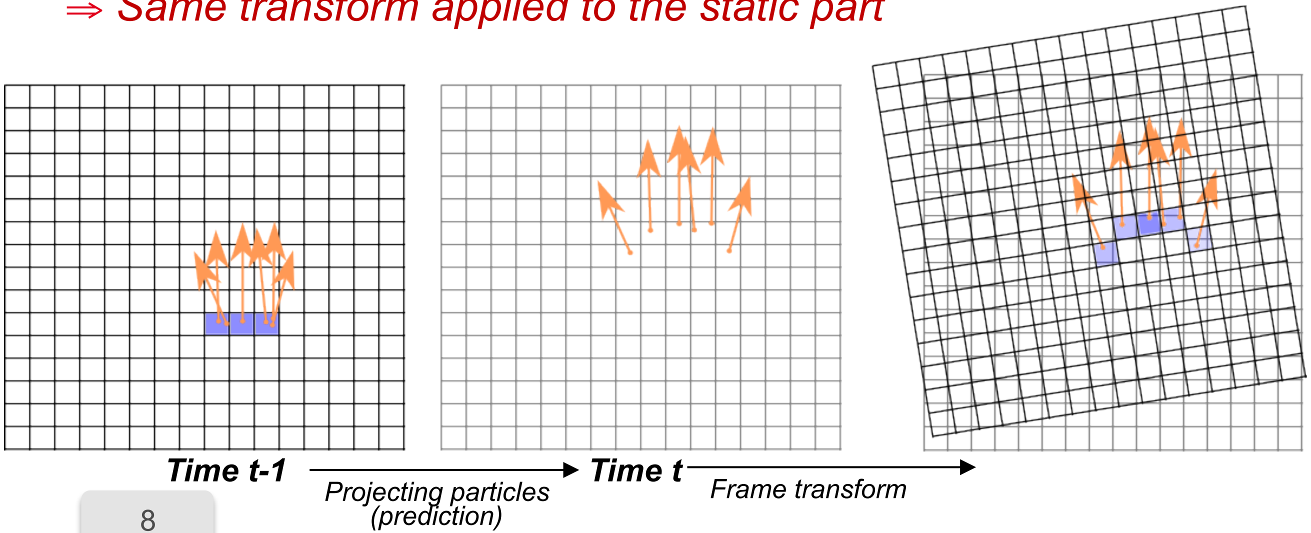
- **Basic idea**: *Modify the representation structure to avoid the previous computational problems*
  - ✓ Making a clear distinction between **Static & Dynamic & Free** components
  - ✓ Modeling velocity using **Particles** *(instead of histogram)*
  - ✓ Making an **adaptive repartition** of those particles in the grid



DP-Grids – *How to compute P(OV | Z C) in practice?*
HSBOF updating process (principle)

- **Introducing a Dynamic model for "projecting" particles in the grid** *($S_{t-1} \rightarrow S_t$)*
  - ⇒ *Immediate antecedent association*
  - ⇒ *Simplified velocity prediction to the cells*

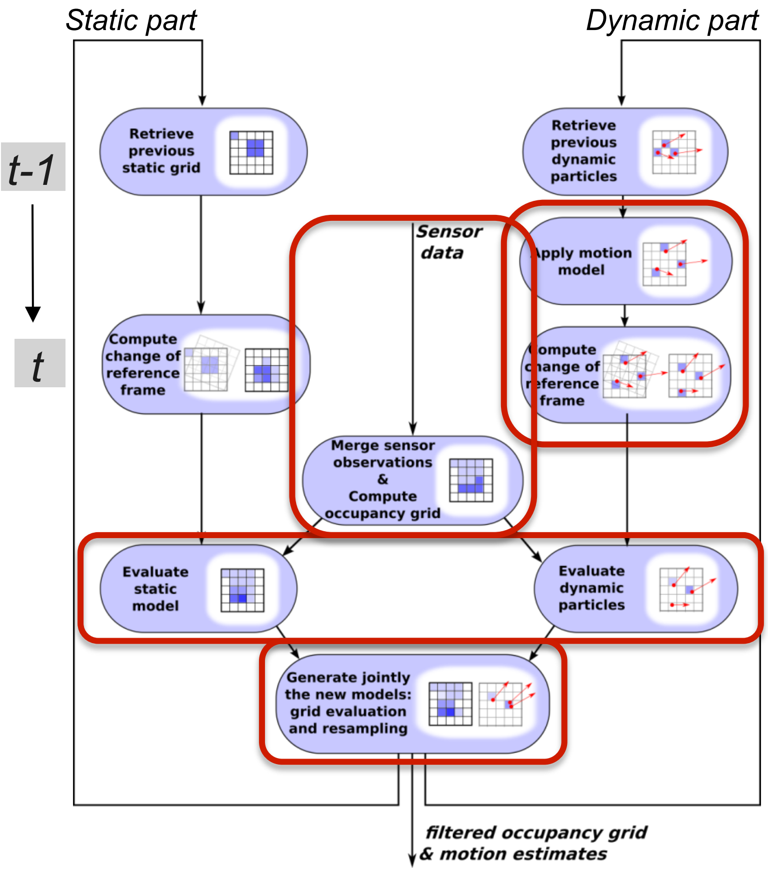
- **Updating Grid Reference Frame**
  - ⇒ *Translation & Rotation values provided by sensors (Odometry + IMU)*
  - ⇒ *Same transform applied to the static part*



DP-Grids – *How to compute P(OV | Z C) in practice?*
HSBOF updating process (outline)

Important steps in the updating process

- Dynamic part (particles) is "**projected**" in the grid using motion model *(motion prediction)*
- Both Dynamic & Static parts are expressed in the **new reference frame** *(moving vehicle frame)*
- The two resulting representations are confronted to the **observations** *(estimation step)*
- **New representations (static & dynamic)** are jointly evaluated and particles re-sampled



DP-Grids – *How to compute P(OV | Z C) in practice?*

HSBOF filtering calculation

$$P(O\,V\,|\,Z\,C)=\lambda\sum_{A\,O^{-1}\,V^{-1}}P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

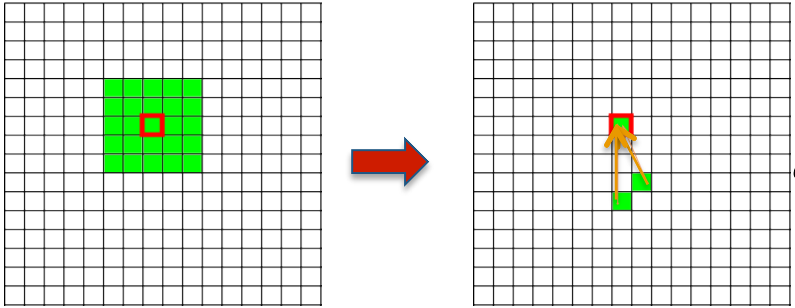
~~Sum over the neighborhood, with a single velocity per antecedent~~

New computation approach

→ *Sum over* ***the particles projected in the cell*** *& their* ***related static parts***

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\,P(O^{-1}\,V^{-1}\,|\,A)\,P(O\,V\,|\,O^{-1}\,V^{-1})\,P(C\,|\,A\,V)\,P(Z\,|\,O\,C)$$

Previous computation approach



New computation approach

HSBOF: Main features

- Empty & Static components ➔ **Occupancy Grid**

- Dynamic components ➔ **Sets of Particles** (Motion field)
  - ✓ *Smooth integration of the ego-motion (IMU & Odometry)*
  - ✓ *Propagation of sets of particles in the Grid*
  - ✓ *Joint estimation of distributions*

- More efficient (computation & memory) & Better estimation of velocities

[Negre et al 2014]

Experimental results in urban environment