

Bootstrapping of Sensor Networks

in Ubiquitous Tracking Environments

Diploma Thesis: Final Presentation

Franz Strasser

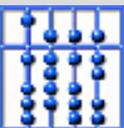
Technische Universität München

strassef@in.tum.de

20.07.2004

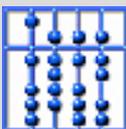
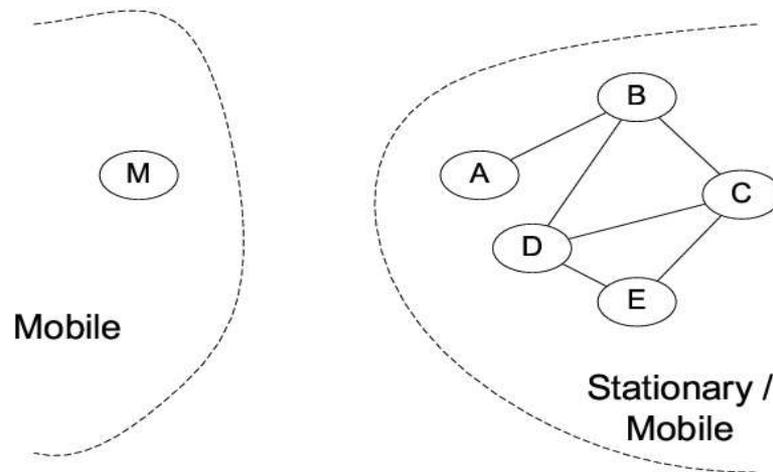
Overview

- Bootstrapping: Integration of mobile sensor set-ups into stationary or mobile ones
 - Scalability in large sensor networks
 - Support for mobile set-ups
- Scenarios should prove the concept for mobile support
 - Mobile set-ups often have only incomplete configuration
 - SR graphs are incomplete because measurements are missing
 - Estimation of static spatial relationships
- Motion pattern analysis
 - Finding similarities in the motion behavior indicates static relationships



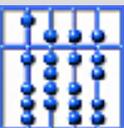
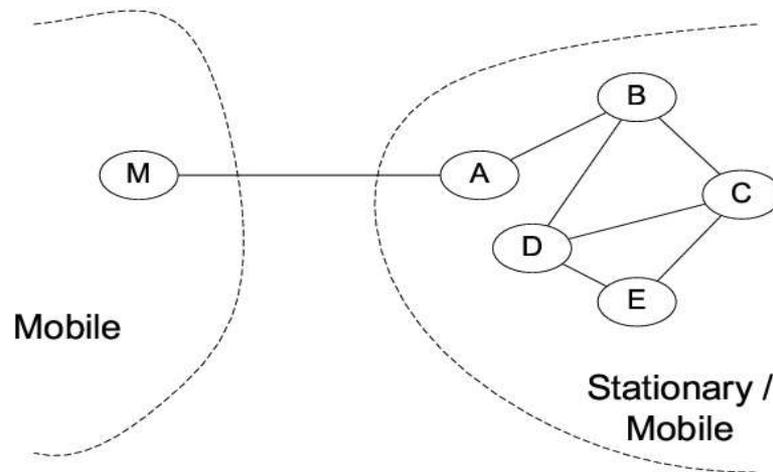
Definition of *Bootstrapping*

- Bootstrapping: setting up a complex system with the help of simpler, smaller parts. Knowledge of the overall system is given by the system itself.
- For Ubitrack: Informally:
Merge two separated SR graphs by adding new measurement edges



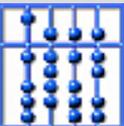
Definition of *Bootstrapping*

- Bootstrapping: setting up a complex system with the help of simpler, smaller parts. Knowledge of the overall system is given by the system itself.
- For Ubitrack: Informally:
Merge two separated SR graphs by adding new measurement edges



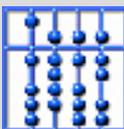
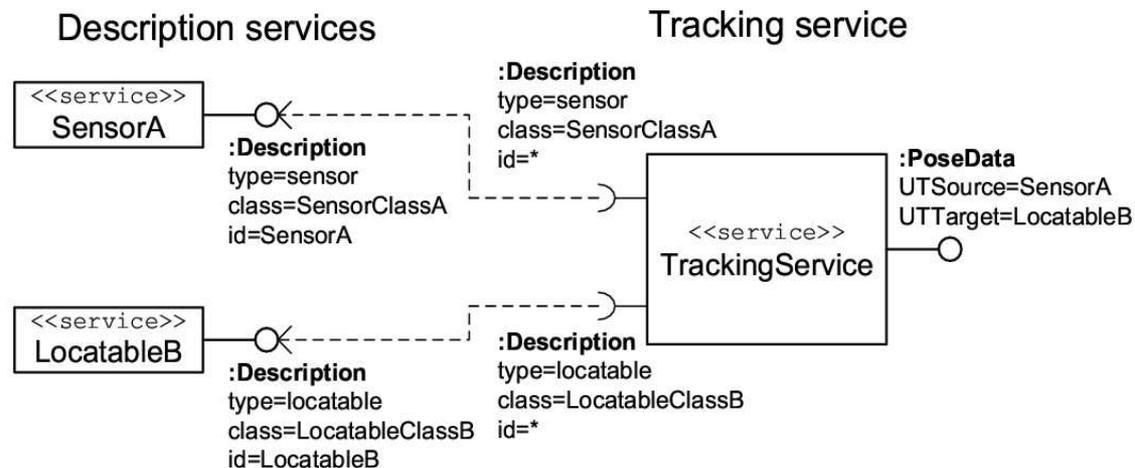
Bootstrapping steps

1. Connect to unknown ambient network infrastructure
 - Use DWARF: Building ad-hoc networks is done by the Service Manager in a suitable way
 - Create communication channels between Ubitrack set-ups
2. Incorporate new sensors at runtime and create new measurements
 - Exchange configuration information in advance
 - Deducing spatial relationships by using all available measurements
3. Merge the separated Spatial Relationship Graphs by adding the new relationships as edges
 - UMAs must update their SR graphs to consider these new edges for their graph search



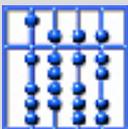
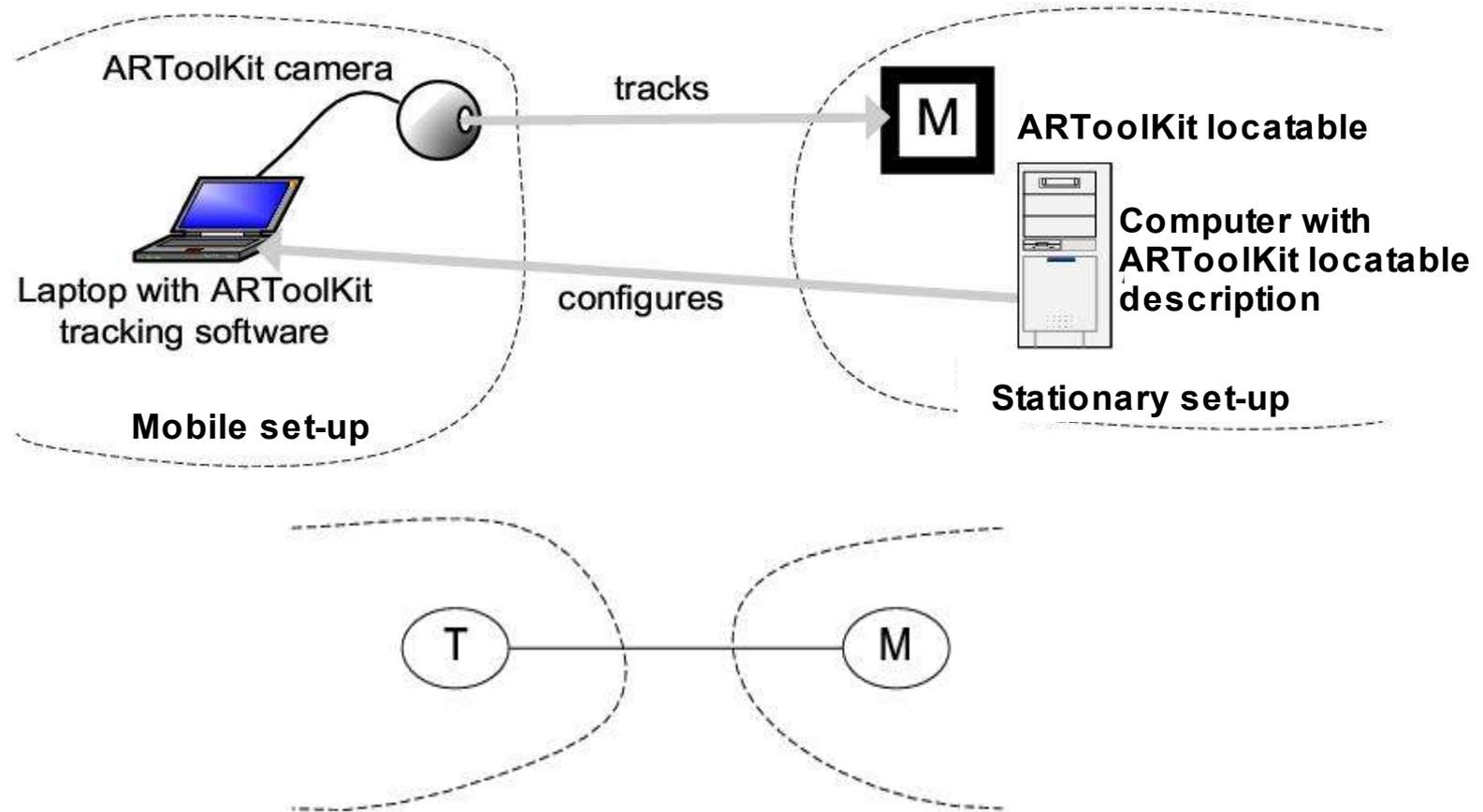
Exchange of Configuration Information

- SR Edges are measurements
 - Produced by components of data flow graphs, especially by sensors
 - Realized in Ubitrack by tracking services
- Tracking services may need configuration to supply measurements

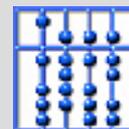
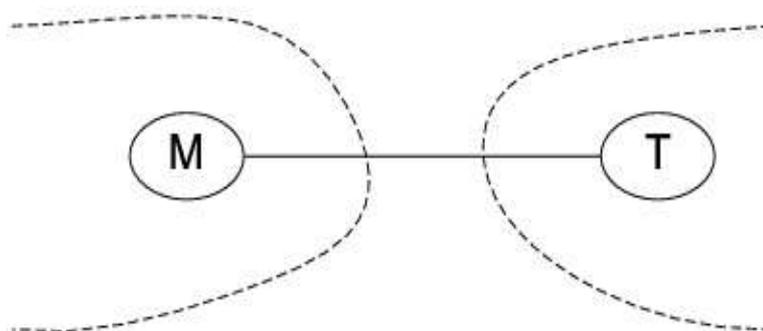
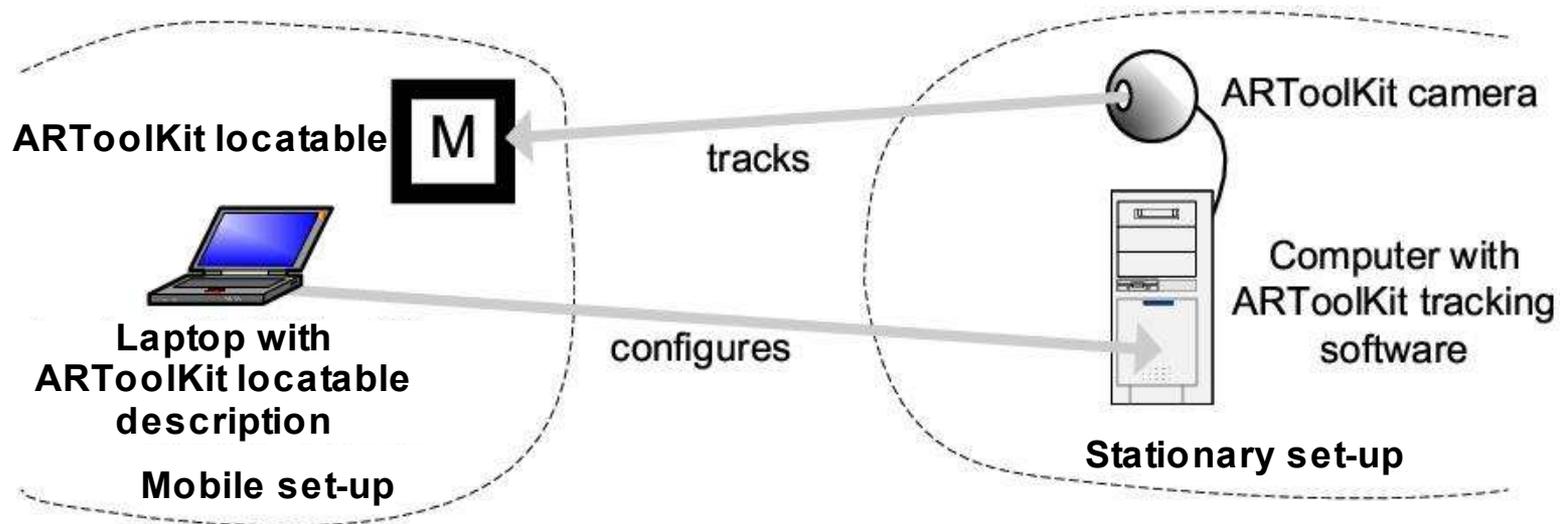


Mobile Scenarios

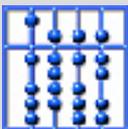
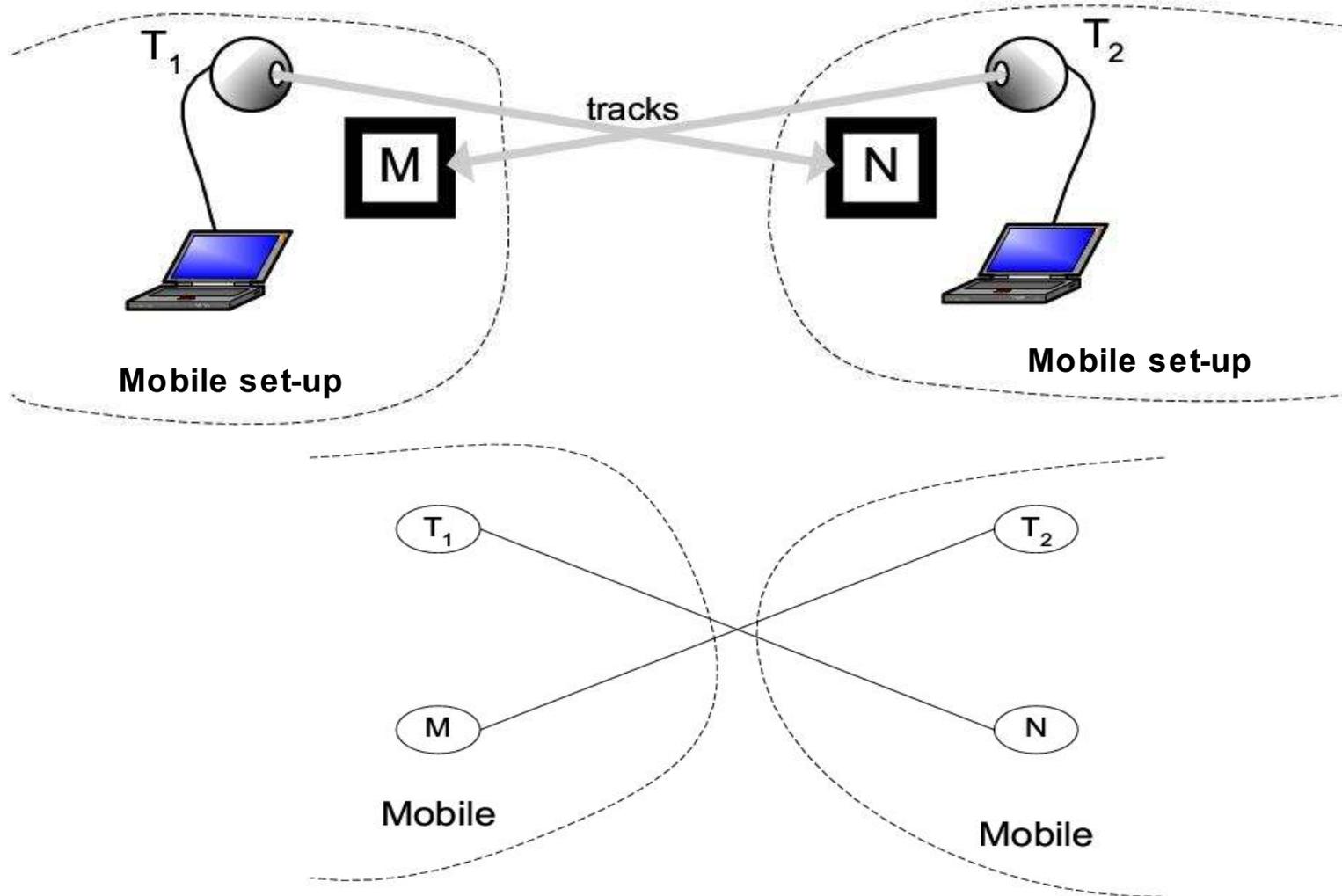
- Incorporate new sensors: scenarios with a mobile set-up
- Scenario 1:



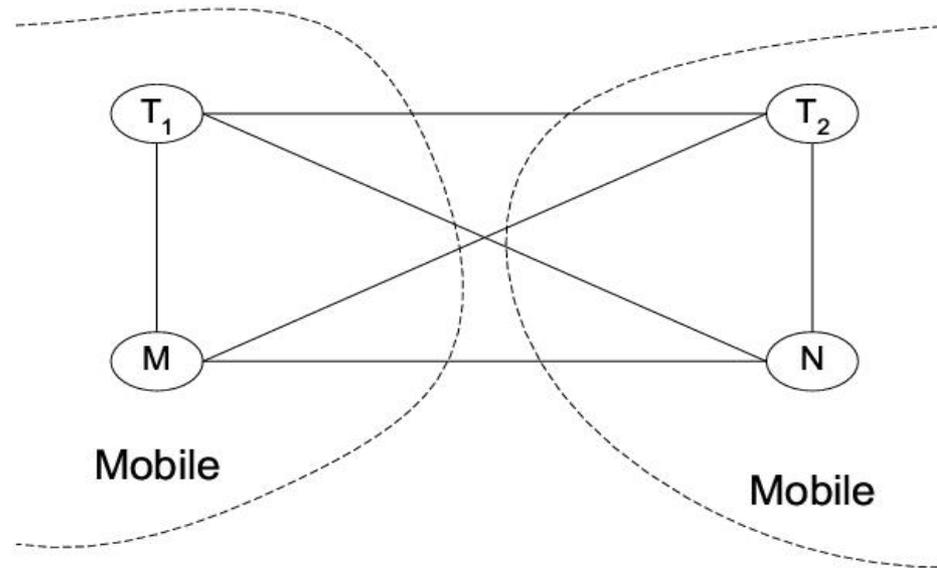
Scenario 2



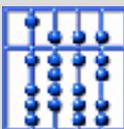
Scenario 3



Scenario 3: Fully Connected SR Graph

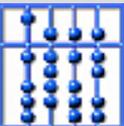


- This is the desired situation. Why?
 - Between every node exist at least a second path
 - Fusion of the results => Improved accuracy
 - We can use the second measurement to improve the first
- Need another bootstrapping step



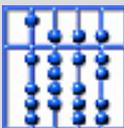
Next BS step: Deduce new relationships

- How can new relationships be estimated?
 - No measurements are available
 - Finding static relationships between objects
 - No common frame of reference: no common coordinate system
- Finding similarities in the motion behavior
 - Observe dynamic quantities over a certain time range:
 - Velocity: Speed, angular velocity
 - Acceleration
 - Why? Because no coordinate system: cannot compute transformation matrix
 - Motion pattern analysis
 - Direct comparison
 - Frequency analysis



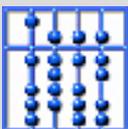
Frequency Analysis of Motion Patterns

- Theory:
 - Observe the PoseData for a certain time interval and compute the speed and the angular velocity
 - Compute the FFT for these two time dependent signals
 - Compute the power spectrum of one signal: amplitudes of the frequencies
 - Compare two signals by using the coherence function
 - Compute a propability value for a static relationship
- The propability value indicates if the two measurements track similar motion behavior
- Human motion: between 0 and 10Hz



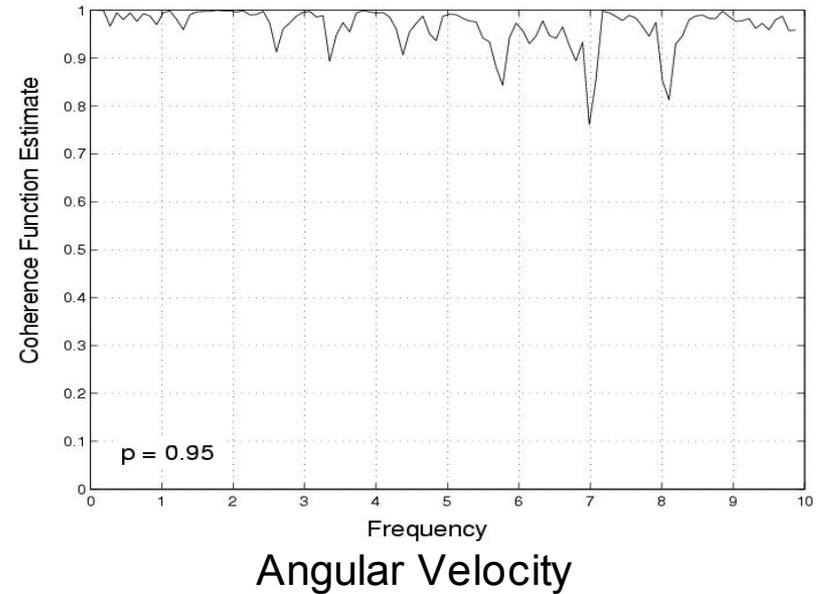
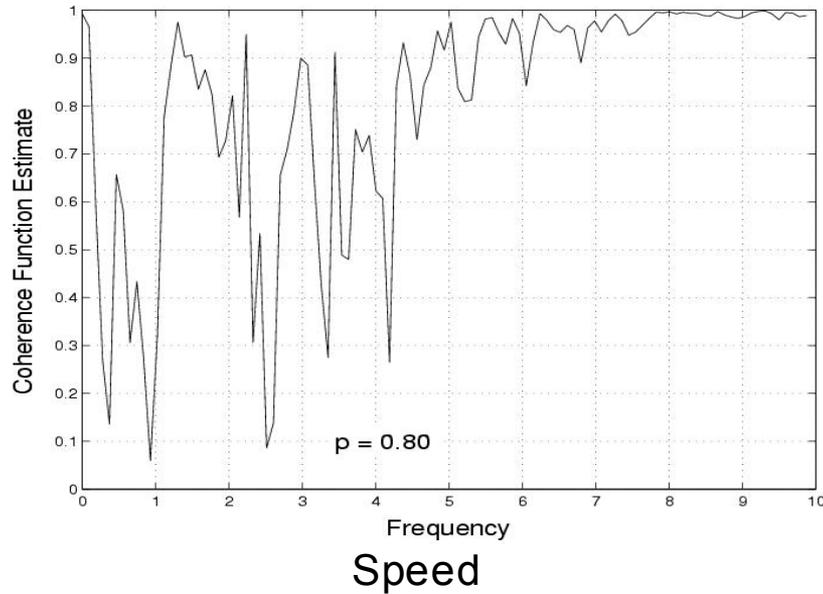
Test Cases

- Tested the approach if suitable for the common sensors in the AR lab: ART, ARToolKit, Intersense Intertrax
- 5 Test case: Offline analysis with MATLAB
- Every test case produced following results:
 - Velocity curves: speed and angular velocity
 - Coherence curve for both
 - Propability values for both



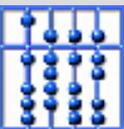
Results 1/4

- Attached ART targets



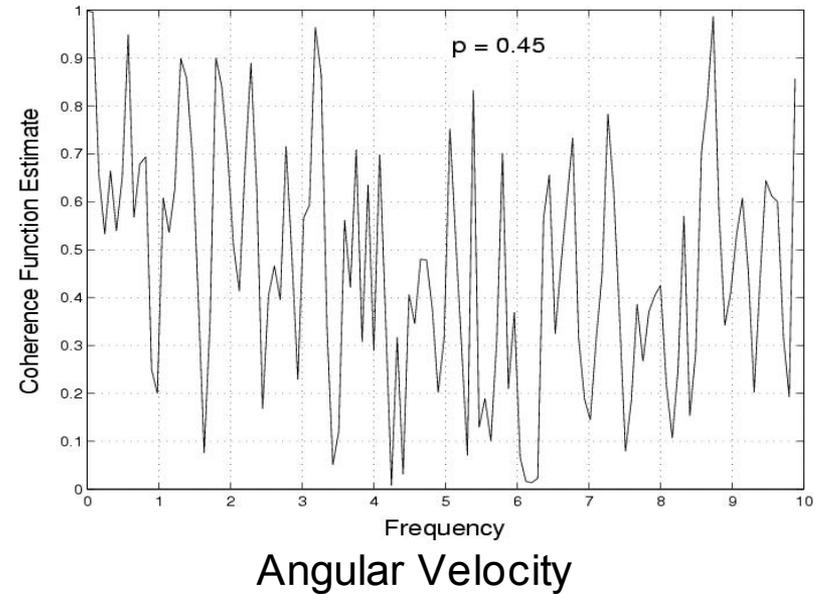
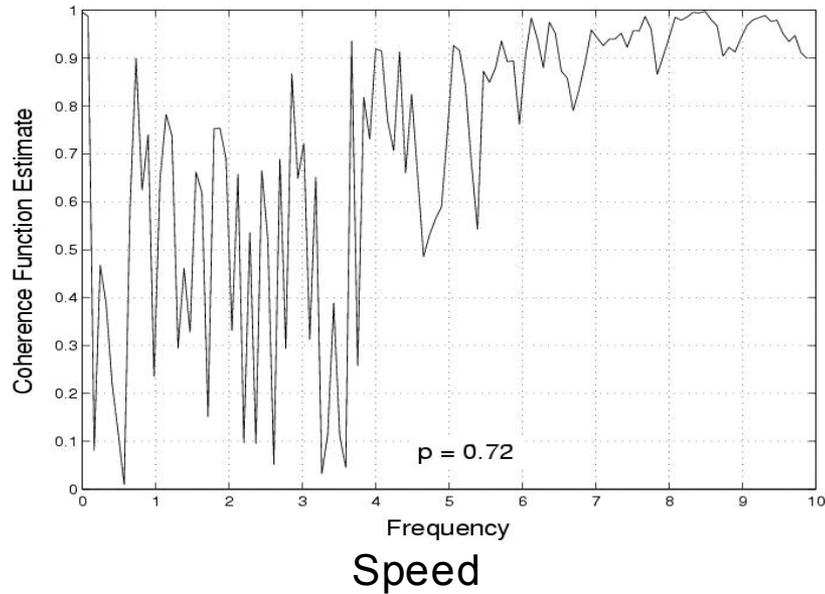
Result:

- Works
- Speed: relatively poor result
- Angular velocity is better



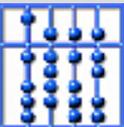
Results 2/4

- Detached ART targets



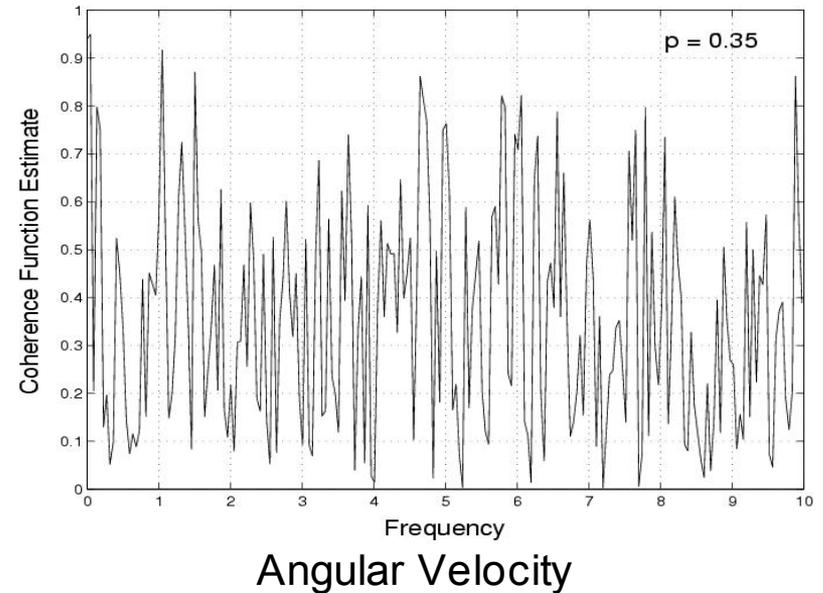
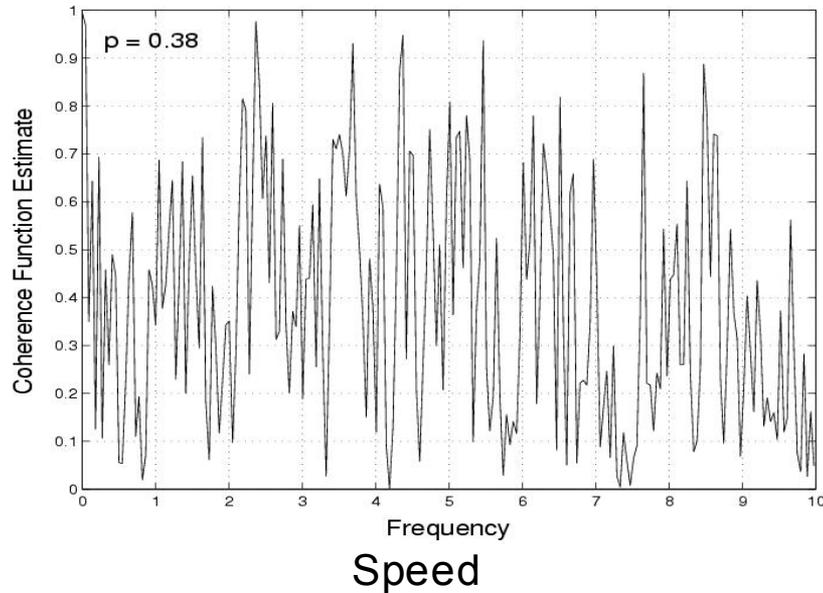
Result:

- Works
- Speed: relatively poor result
- Angular velocity is better



Results 3/4

- ART + ARToolKit

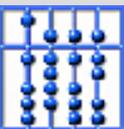


Result:

- Does not work, even with Kalman Filter
- Both show uncorrelated behavior

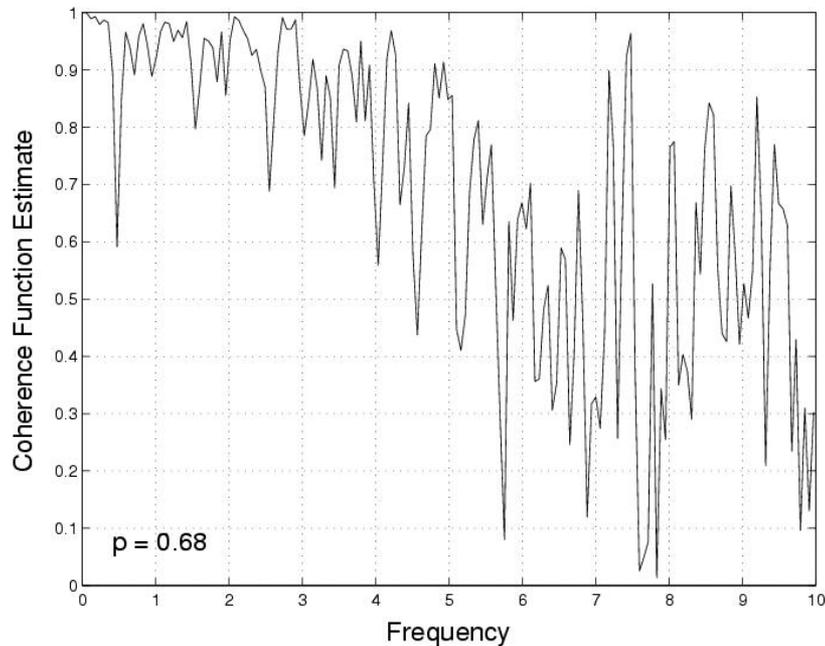
Why?

- Outliers: used a simple removal algorithm
- Time offset in the measurements



Results 4/4

- ART + Intertrax



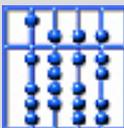
Angular Velocity

Result:

- Works
- Coherence is high in lower frequencies (human motion)
- Probability value is low, but coherence differs only in high frequencies:
Indication for sensor noise

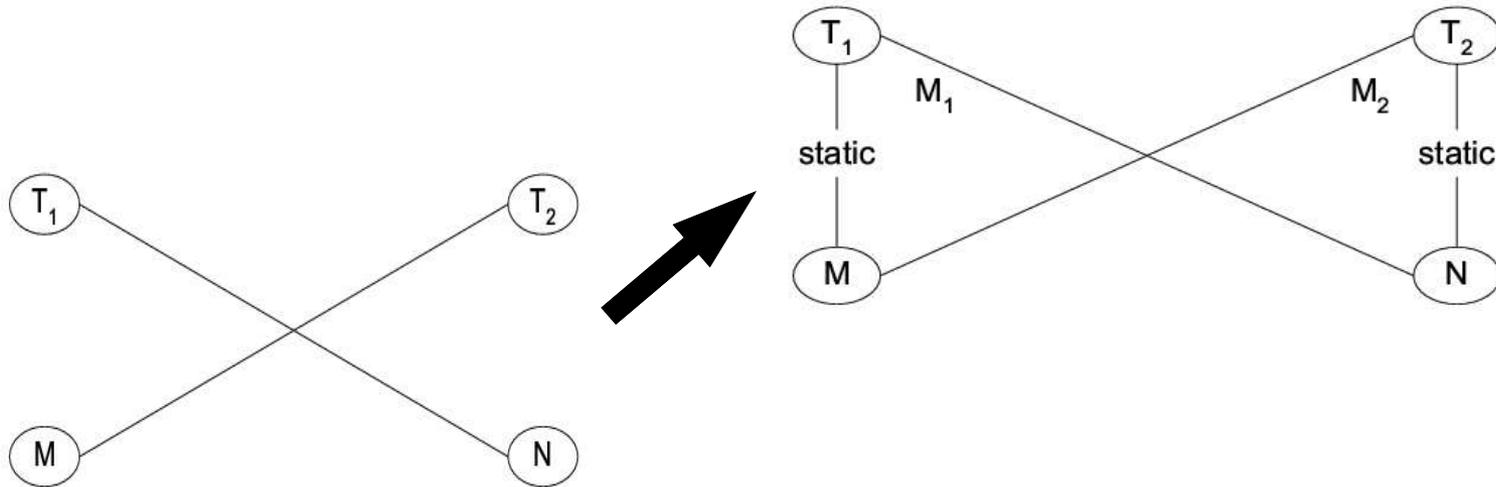
Improvements:

- Weigh low frequencies more than higher ones



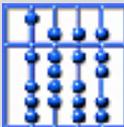
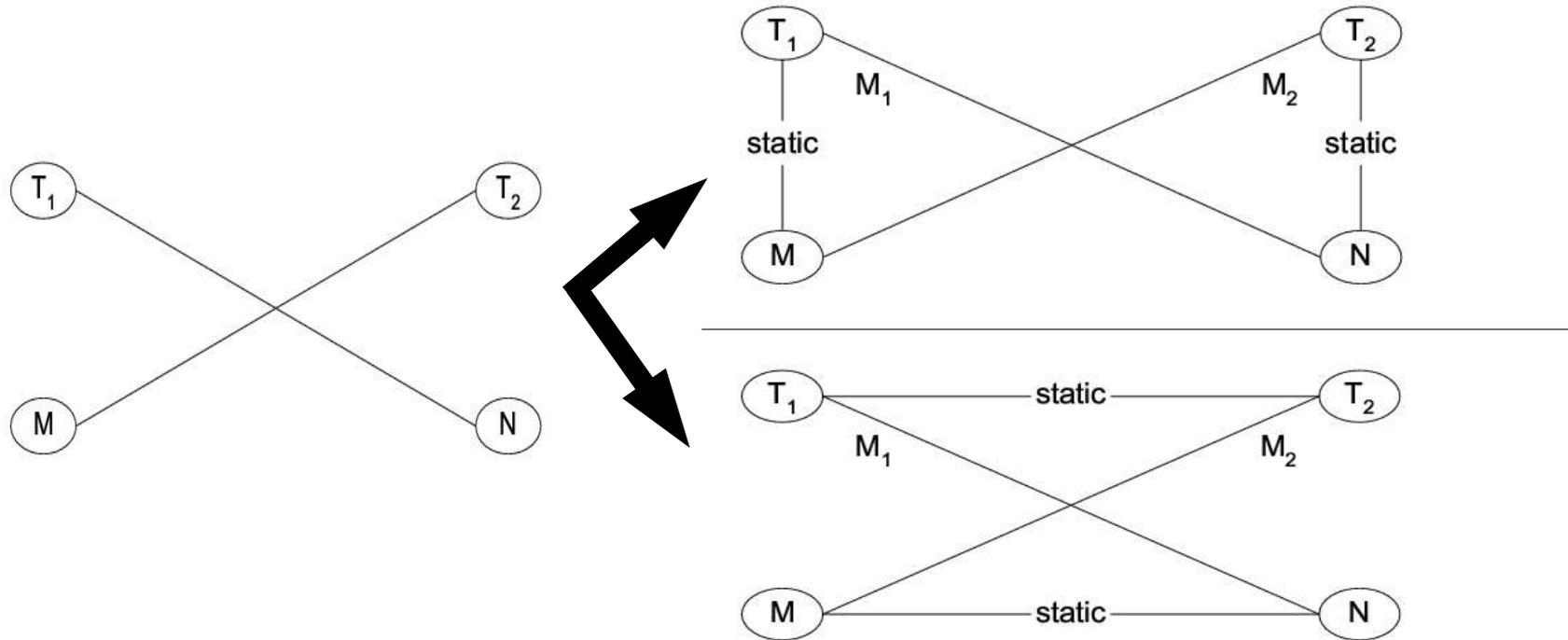
Merging of SR Graphs

- Now back to Bootstrapping:
 - Use the results of the analysis to add new relationships
- However: Ambiguity !!



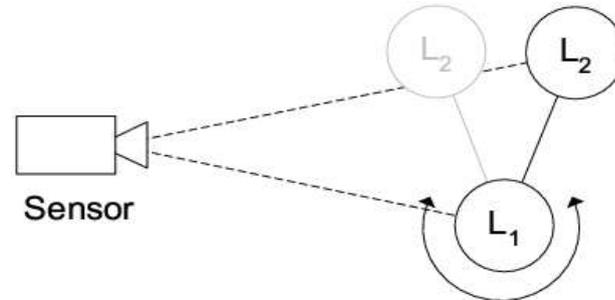
Merging of SR Graphs

- Now back to Bootstrapping:
 - Use the results of the analysis to add new relationships
- However: Ambiguity !!

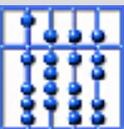


Conclusions

- Angular velocity is a better criteria to determine static relationships:

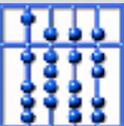


- Original approach was used on accelerometers
 - Approach also works with speed and angular velocity
- Future work
 - Solving the ambiguity:
 - Additional information from the mobile set-ups, e.g. their static relationships
 - Assume one case and prove or negate it by long-term observations



Acknowledgements

- Thanks to my advisors: Martin Wagner, Asa MacWilliams and Martin Bauer !
 - Thanks to Gudrun Klinker !
 - Thanks to Dagmar and Daniel !
 - Thank YOU for your attention !!
-
- Any Questions?



Scenario 3b

