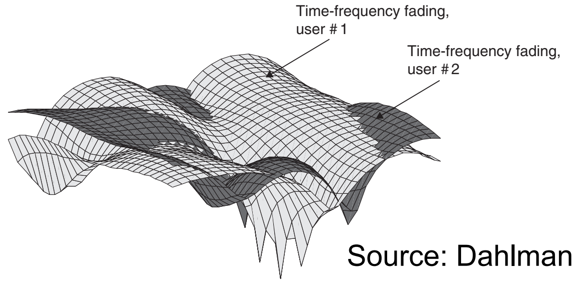
Technische Universität München
Lehrstuhl für Kommunikationsnetze
Prof. Dr.-Ing. J. Eberspächer

Simulation and Prototyping Workshop
RWTH Aachen, July 14, 2011

IMTAphy: An IMT-Advanced Spatial Channel Model Implementation and PHY-Abstraction for System-Level Simulations

Jan Ellenbeck
jan.ellenbeck@tum.de

Motivation



Source: Dahlman et al.

- LTE, WiMAX, and future IMT-A cellular systems exploit
  - time/frequency/spatial selectivity of the channel by fast scheduling with link adaptation
    ➔ need multi-path / fast-fading MIMO channel model

- Our general research topic: interference management (IM)
  - IM restricts scheduler‘s degrees of freedom (frequency reuse, usage patterns in time, available beamforming / precoding)

  ➔ to judge tradeoff of IM schemes: fast-fading spatial channel model needed

- ITU-R M.2135 mandates system-level simulations to evaluate mean cell / cell-edge spectral efficiency of IMT-A candidate systems
  - need an efficient implementation
  - that gives accurate / reliable results

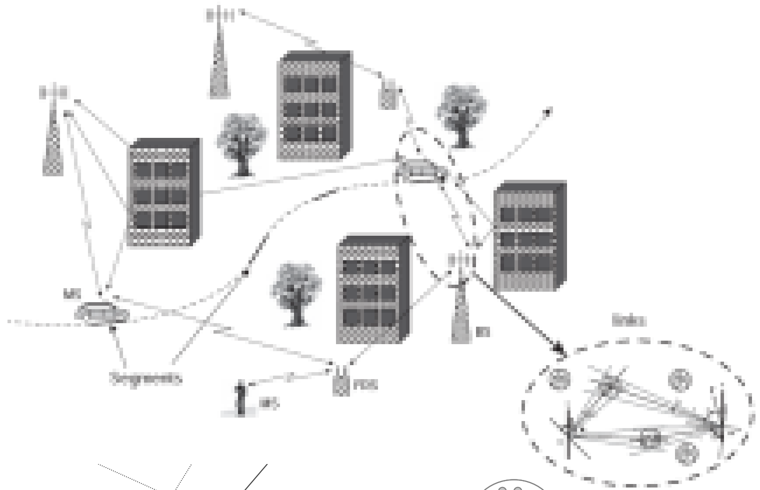
➔ C++ implementation of the M.2135 channel model available under GPLv3 license at https://launchpad.net/imtaphy

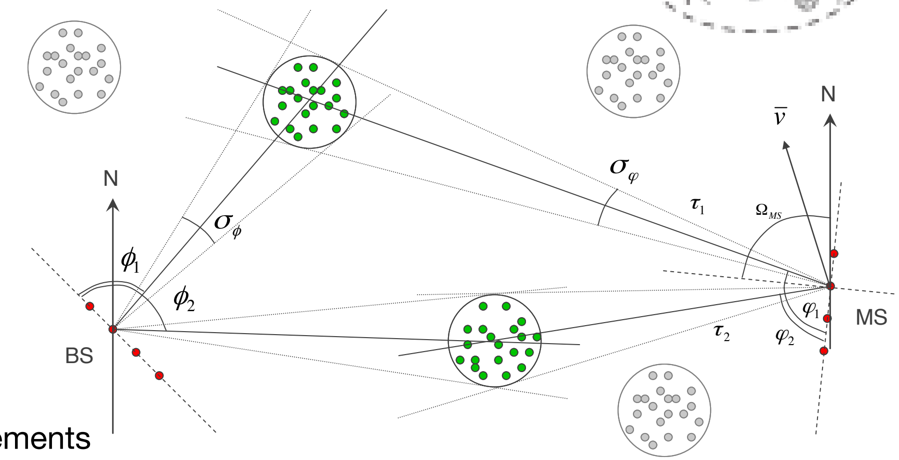
➔ PHY abstraction (MIMO receivers, eff. SINR, BLERs) and simple LTE implementation will be available shortly

ITU-R IMT-A Channel Model (Primary Module)

- Adapted from WINNER+ channel model
- Link-level and system-level simulations possible

- Large Scale effects:
  - pathloss
  - outdoor shadowing
  - penetration shadowing
    Outdoor-to-Vehicle (O2V), Outdoor-to-Indoor (O2I)
  - antenna patterns
  - handover margin
  - feeder loss
- Small Scale effects (due to multi-path) :
  - geometry based stochastic model
  - channel impulse response (CIR) with proper
    - angular power distribution among paths
    - power delay profile
    - phase relationships between antenna array elements
  - correlations between:
    - antenna elements
    - mobiles associated to same base station



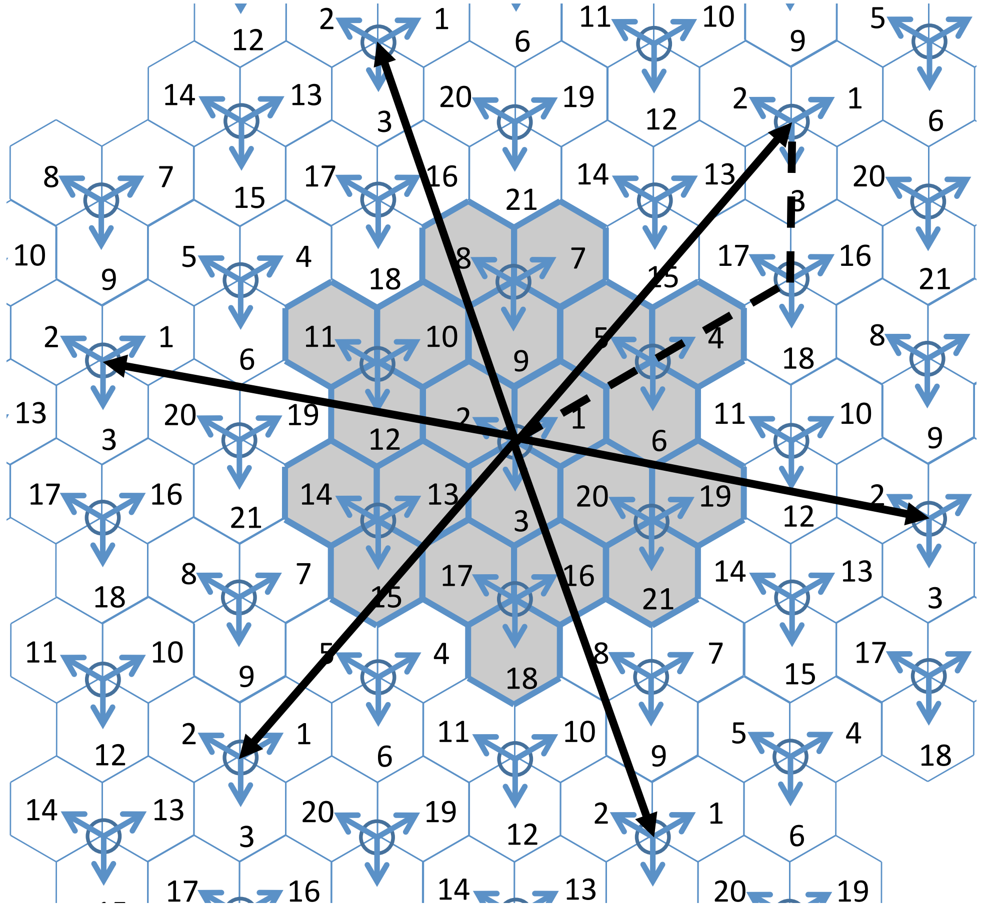


Source: IMT-Advanced Evaluation Guidelines ITU-R M.2135

Wraparound Simulation for Hexagonal Scenarios

Problem:

- M2135: explicit interference modeling
- Same (full) protocol stack in all nodes for system-level simulation (complexity!)
- But realistic interference situation only in inner cells

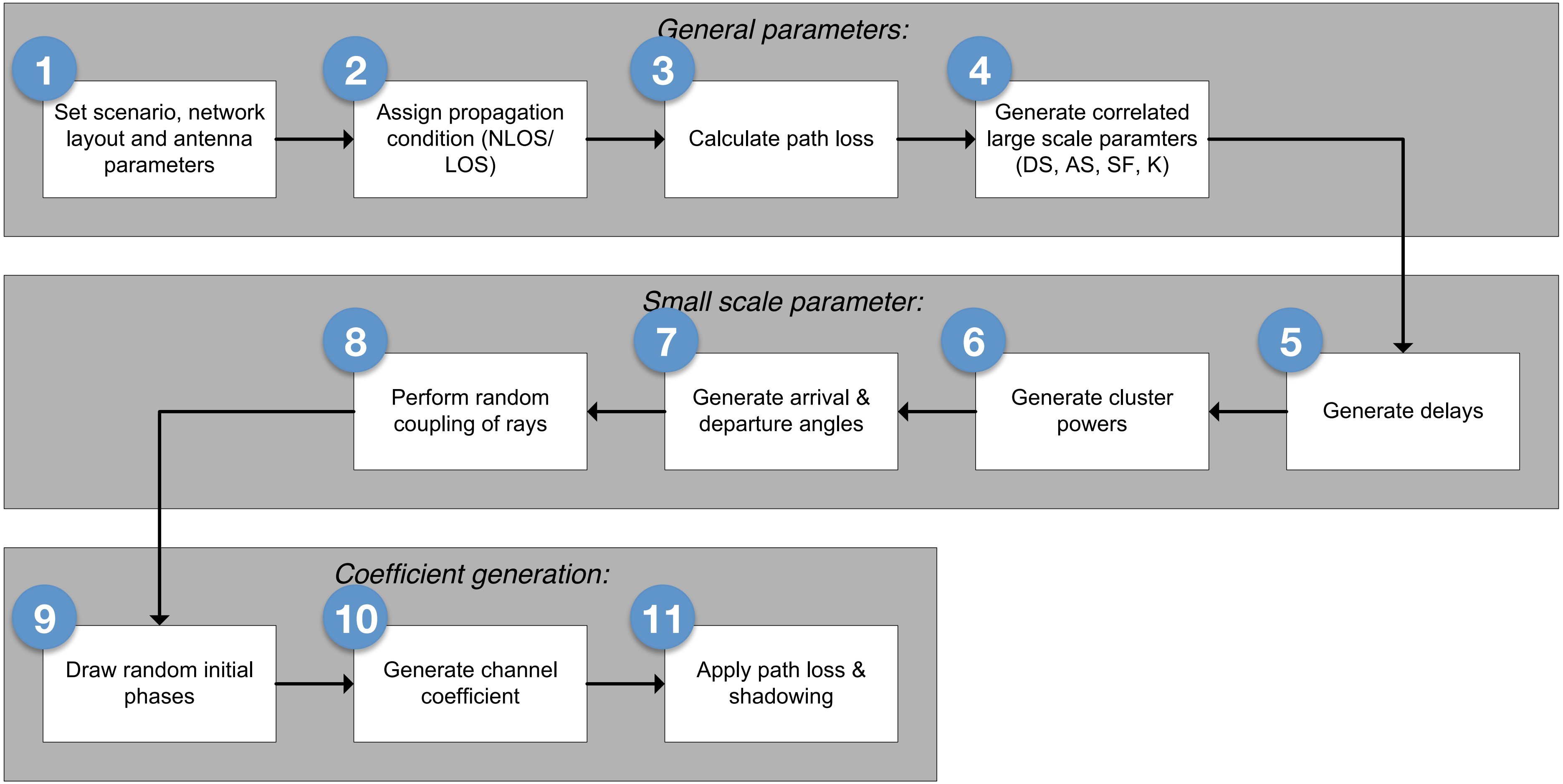


Solution: wraparound

- makes cells at opposing ends neighbors (like on a torus)
- here: per link shift mobile to all possible wraparound positions and choose closest

➔ All nodes give realistic statistics

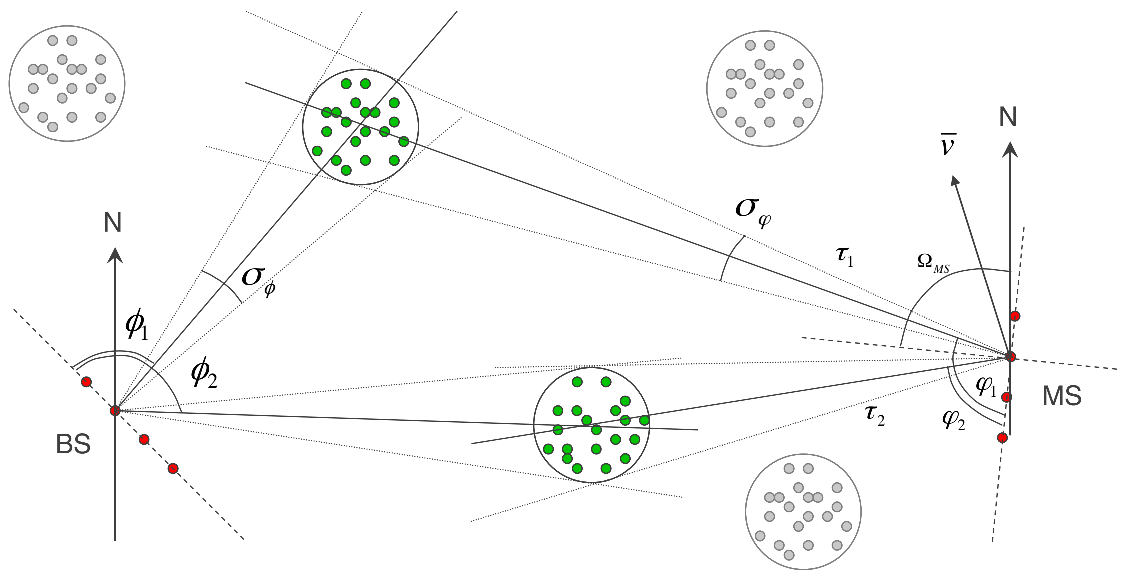
Channel Coefficient Generation Procedure



Source: Rep. ITU-R M.2135

Step 10: Generate Channel Coefficients

$$h_{u,s,n}(t) = \sqrt{P_n} \sum_{m=1}^{M} \begin{pmatrix} F_{rx,u,V}(\varphi_{n,m}) \\ F_{rx,u,H}(\varphi_{n,m}) \end{pmatrix}^T \begin{pmatrix} \exp(j\Phi_{n,m}^{vv}) & \sqrt{\kappa^{-1}}\exp(j\Phi_{n,m}^{vh}) \\ \sqrt{\kappa^{-1}}\exp(j\Phi_{n,m}^{hv}) & \exp(j\Phi_{n,m}^{hh}) \end{pmatrix} \begin{pmatrix} F_{tx,s,V}(\phi_{n,m}) \\ F_{tx,s,H}(\phi_{n,m}) \end{pmatrix}$$
$$\exp(jd_s 2\pi\lambda_o^{-1}\sin(\phi_{n,m}))\exp(jd_u 2\pi\lambda_o^{-1}\sin(\varphi_{n,m}))$$
$$\exp(j2\pi\upsilon_{n,m}t)$$



$H_{u,s,n}(t)$	channel coefficient
u	number of Rx antennas
s	number of Tx antennas
$n = 1 \ldots 24$	cluster (path) index
$m = 1 \ldots 20$	ray index
τ_n	delay of path n
P_n	power of path n
$F_{rx,u,V}(\varphi_{n,m})$	electrical field pattern
$F_{rx,u,H}(\varphi_{n,m})$	for Rx/Tx antennas with
$F_{tx,s,V}(\phi_{n,m})$	vertical / horizontal
$F_{tx,s,H}(\phi_{n,m}$	polarization
$\Phi_{n,m}^{vv}$, $\Phi_{n,m}^{vh}$ $\Phi_{n,m}^{hv}$, $\Phi_{n,m}^{hh}$	random phases
κ	cross polarization power ratio
d_s,	effective distance of Tx / Rx
d_u	elements s/u to reference element
$\phi_{n,m}$	angle of departure
$\varphi_{n,m}$	angle of arrival
$\upsilon_{n,m}$	Doppler frequency component $\upsilon_{n,m} = \lvert\lvert v \rvert\rvert\cos(\varphi_{n,m} - \theta_v)/\lambda_0$

Source: Rep. ITU-R M.2135

Challenge: Complexity

Typical scenario:

- 57 cells
- 10 mobiles per cell

➔ 57 * 57 * 10 = 32.490 links

- 4 x 4 (Tx x Rx) antenna configuration

➔ 16 antenna pairs per link

- Up to *n* = 1..24 paths (clusters) between an antenna pair

➔ 32.490 * 16 * 24 = 12.476.160 channel coefficients to compute (for one *t*)

- Each path consists of *m* = 1..20 rays (+ LoS), so summation goes over

➔ 249.423.200 rays in total

For each time instant *t*, 250 million values are computed and summed up!

We need a very efficient implementation

to simulate over a significant time span

Key Observation: Most Things are time-invariant!

$$h_{u,s,n}(t) = \sqrt{P_n} \sum_{m=1}^{M} \begin{pmatrix} F_{rx,u,V}(\varphi_{n,m}) \\ F_{rx,u,H}(\varphi_{n,m}) \end{pmatrix}^T \begin{pmatrix} \exp(j\Phi_{n,m}^{vv}) & \sqrt{\kappa^{-1}} \exp(j\Phi_{n,m}^{vh}) \\ \sqrt{\kappa^{-1}} \exp(j\Phi_{n,m}^{hv}) & \exp(j\Phi_{n,m}^{hh}) \end{pmatrix} \begin{pmatrix} F_{tx,s,V}(\phi_{n,m}) \\ F_{tx,s,H}(\phi_{n,m}) \end{pmatrix}$$
$$\exp(j d_s 2\pi \lambda_o^{-1} \sin(\phi_{n,m})) \exp(j d_u 2\pi \lambda_o^{-1} \sin(\varphi_{n,m})) \exp(j 2\pi v_{n,m} t)$$

- Between subsequent time instances, only the red term changes: $\exp(j2\pi v_{n,m}t)$
- The rest (green part) only has to be computed once

Implementation:

1. generate all input parameters with proper correlations
2. pre-compute green part and the coefficient of the red part (without *t)*
3. store for the rest of the simulation (size: e.g. 250 million values)

➔ store with single precision in large vectors to exploit vectorization/caches in CPU

To compute CIR *H* for a new time instant *t*:

1. multiply coefficient with current *t*
2. take complex exponential of (1) (specialized library functions, e.g. Intel MKL vzCIS)
3. multiply constant (green part) with result (2) and sum over all rays (dot product)

Performance Comparison

Computing the CIR for 32,490 links, 4x4 MIMO, 100 TTIs

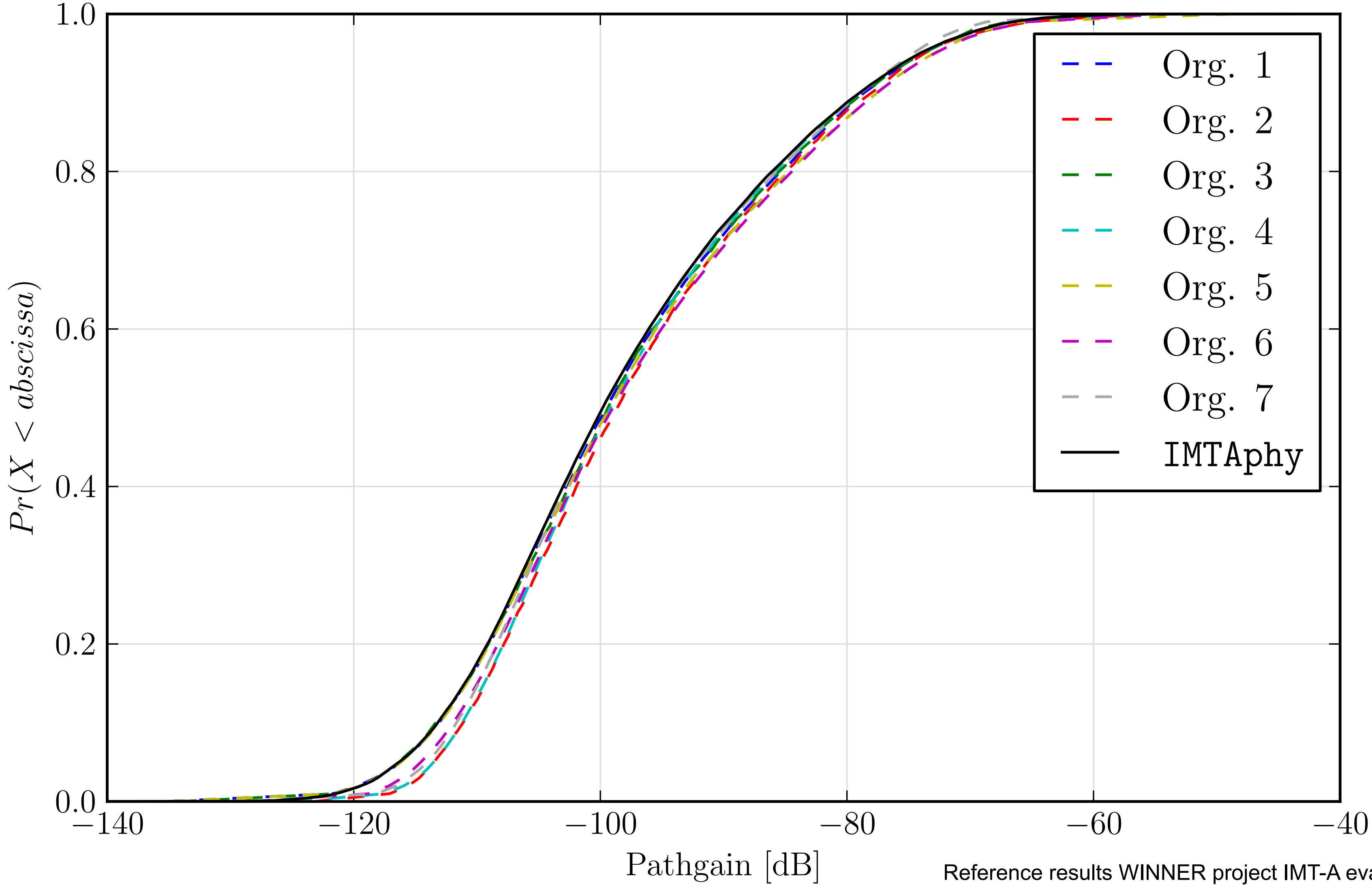
Hardware: 8 cores Intel X5460 @ 3.16GHz (from 2008)

Implementation / IMTAphy mode	Total (real) runtime	Time for each additional TTI	Max. mem. consumption
TTA PG707 [1]	31000 sec.	310 sec.	23 GB
WINNER [2] with C-MEX	1962 sec.	18 sec.	27 GB
IMTAphy 1 thread double 8 threads	1015 sec 501 sec	7.73 sec 4.40 sec	6.4 GB
IMTAphy 1 thread single 8 threads	728 sec 275 sec	4.19 sec 2.21 sec	3.2 GB

[1] TTA PG707: C-Code for Report ITU-R M.2135 Channel Model Implementation http://www.itu.int/oth/R0A06000024/en

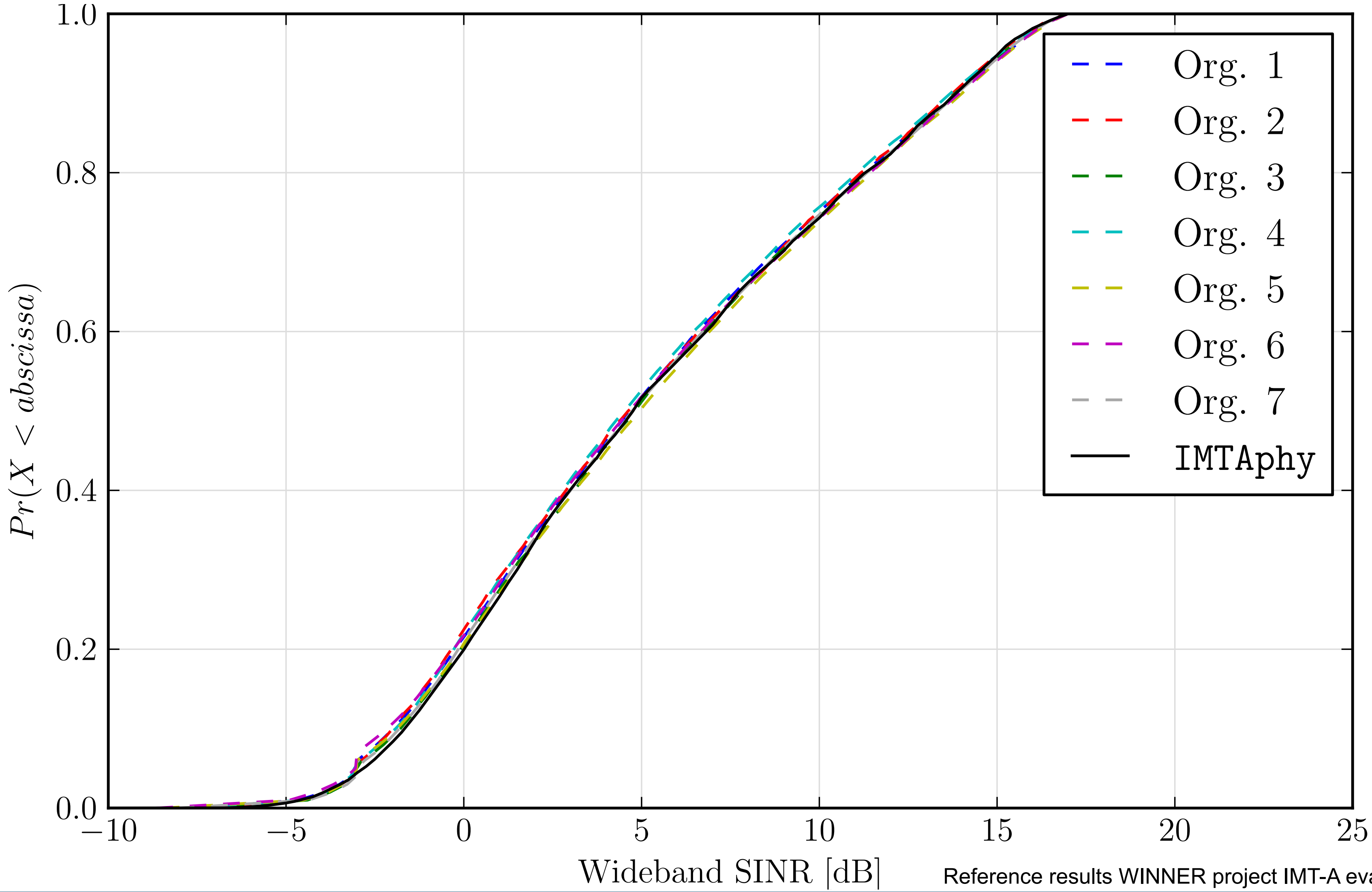
[2] Finland: Software Implementation of IMT.EVAL Channel Model http://www.itu.int/oth/R0A06000022/en

Pathgain Calibration UMa



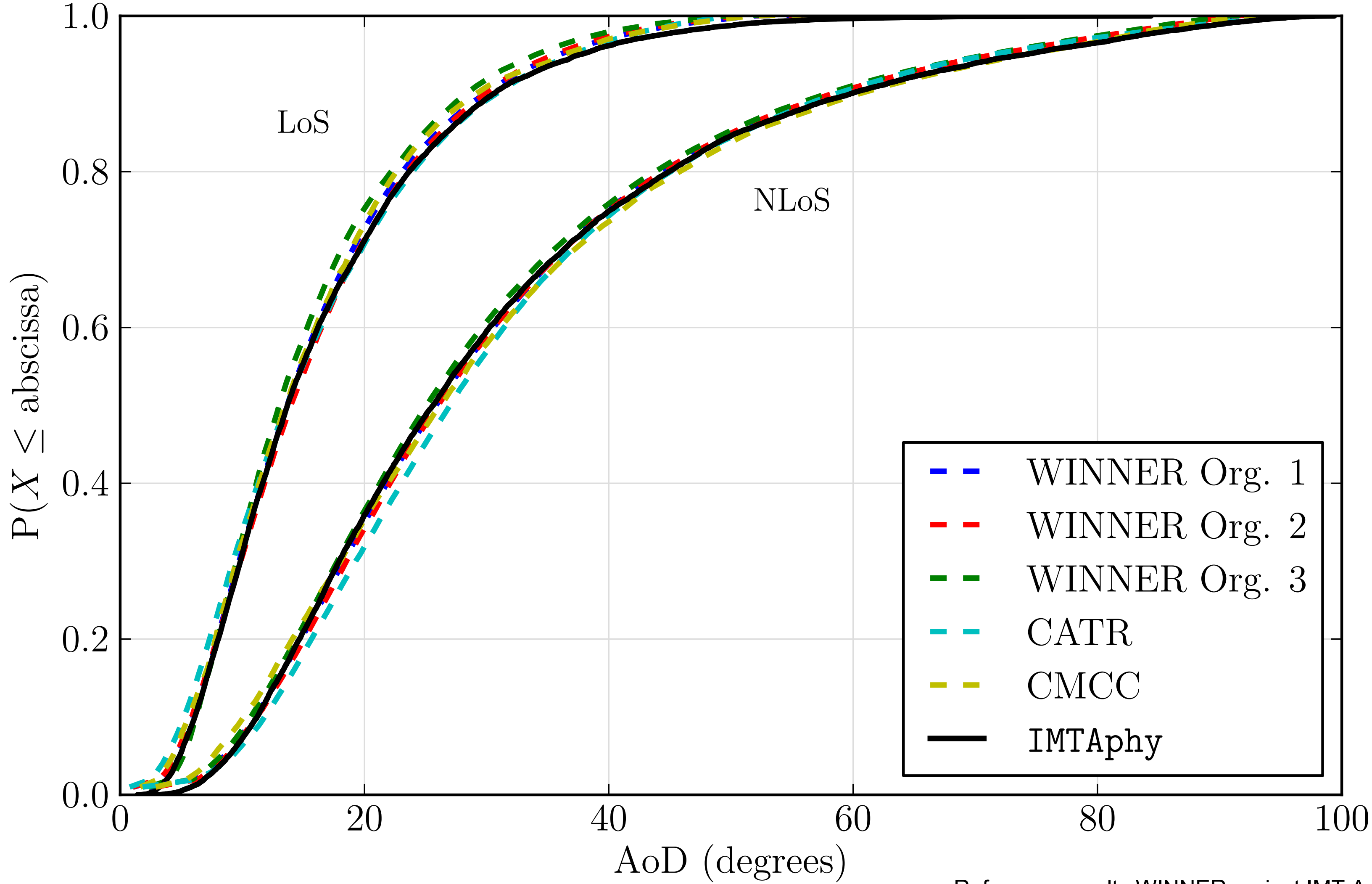
Reference results WINNER project IMT-A evaluation

SINR Calibration UMa



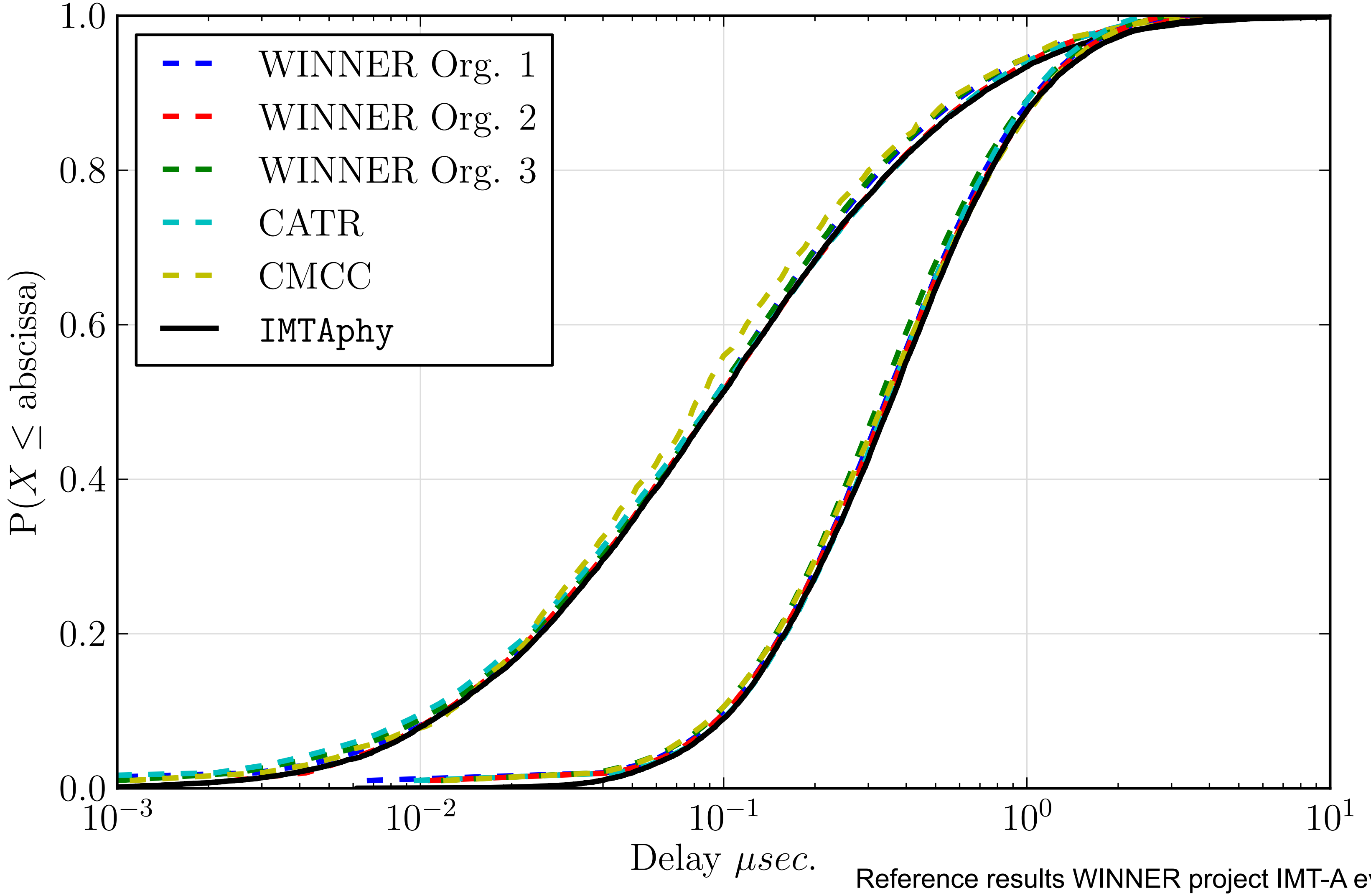
Reference results WINNER project IMT-A evaluation

CDF of Angular Spread at BS (AoD, UMa)



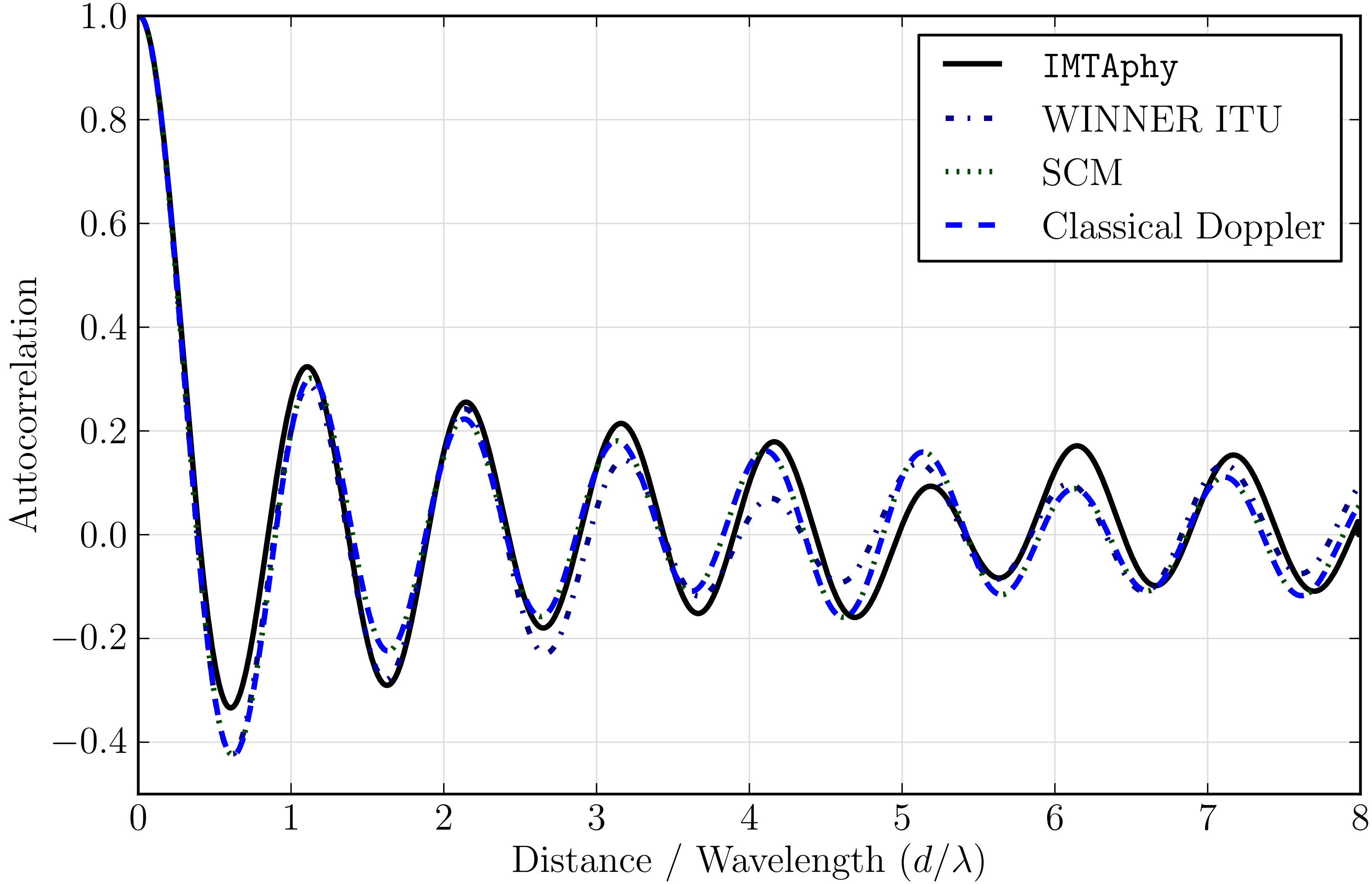
Reference results WINNER project IMT-A evaluation

CDF of Delay Spread UMa scenario



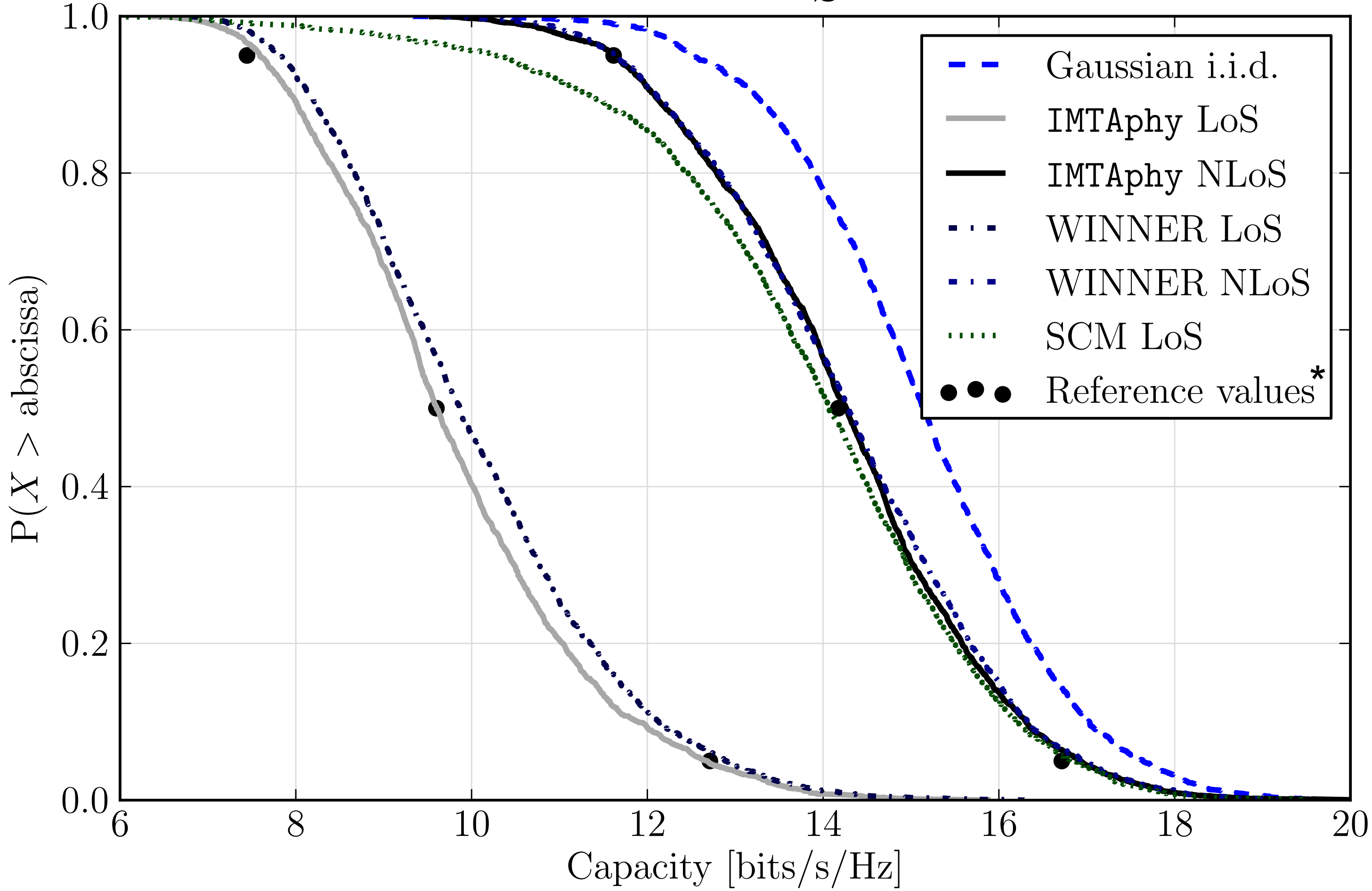
Reference results WINNER project IMT-A evaluation

Temporal Autocorrelation (UMa scenario)



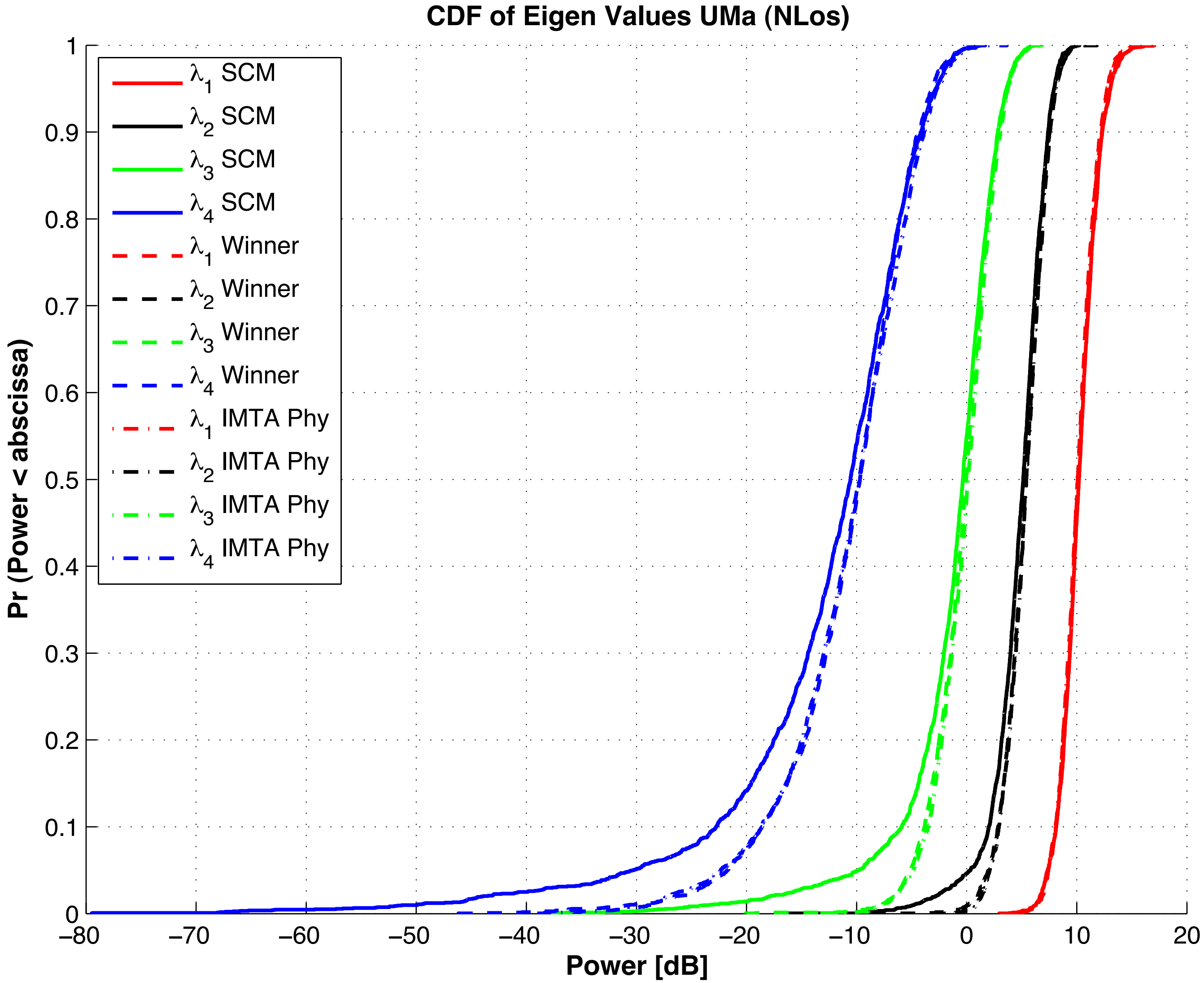
4x4 MIMO Capacity at 14 dB SINR (UMa)

$$C = \log_2 \det \left(\mathbf{I} + \frac{\rho}{S} \mathbf{H}\mathbf{H}^H \right)$$



* Chong et al. "Evolution trends of wireless MIMO channel modeling towards IMT-Advanced," IEICE Trans. on Comm., vol. 92, no. 9, 2009.

Eigenvalue Distribution (UMa NLoS)



Summary M.2135 Spatial Channel Model

- ITU's IMT-Advanced channel model is complex to implement
- Efficient implementations needed for system-level simulation
- Simulators need to be calibrated
- Complete C++ source code together with simulation scenarios and MATLAB evaluation scripts available online under GPL license at: https://launchpad.net/imtaphy

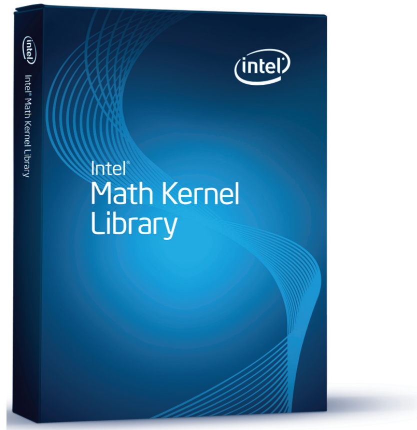
➔ **Reproducible simulation results**

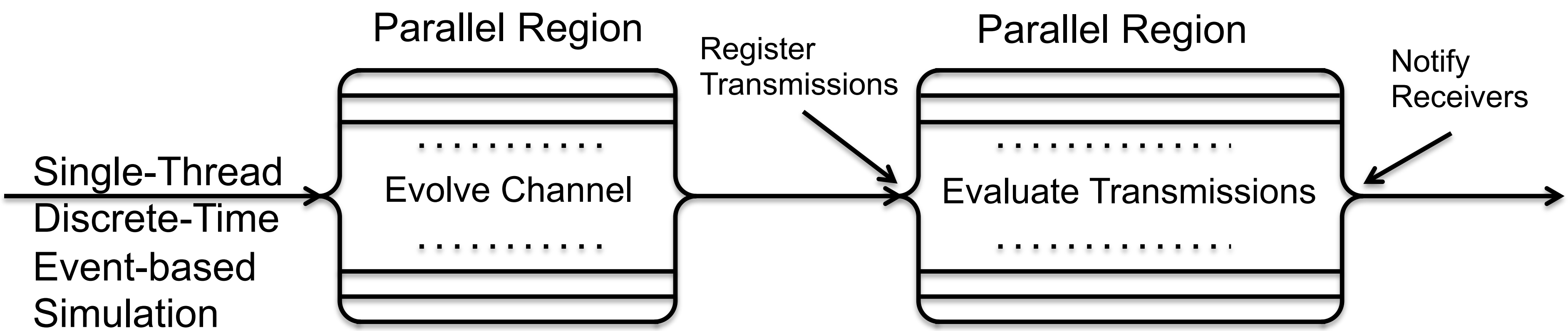
Current LTE Simulator implementation (DL only)

- Phy Abstraction:
  - MIMO (MMSE, MRC) receiver performance model
  - MI-effective SINR computation model (802.16m EMD)
  - Block Error Rate modeling (link level simulations using TU Vienna simulator)
- LTE (Release 8) feedback computation:
  - Derive CQI values from estimated SINRs taking out-of-cell interfering transmissions into account
  - Exhaustive search over closed-loop spatial multiplexing codebook to determine PMI and Rank indicators
- Hybrid-ARQ retransmission support
  - Timing and feedback support (magic)
  - Soft-combining support (currently Chase Combining only)
- Scheduler
  - Currently Round-Robin frequency-nonselective for calibration
  - MU MIMO / spatial multiplexing also supported by IMTAphy
- RLC functionality taken from http:/launchpad.net/openwns-lte

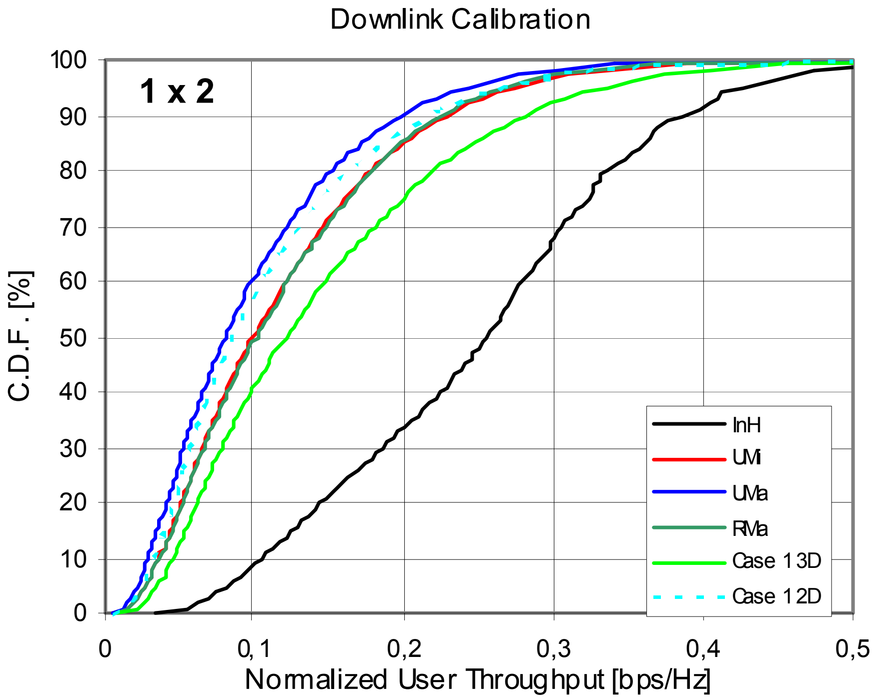
Parallelization

- Main simulation remains single threaded
- Performance critical parts are parallelized
- Time evolution of channel model:
  - OpenMP #pragma omp parallel for
  - Intel Math Kernel Library / Vector Math Library
- Evaluation of transmissions is parallelized:
  - Each TTI, all transmissions register at single entity
  - Afterwards they can be evaluated independently in parallel
  - The results (SINRs) are then serially fed to each Rx node

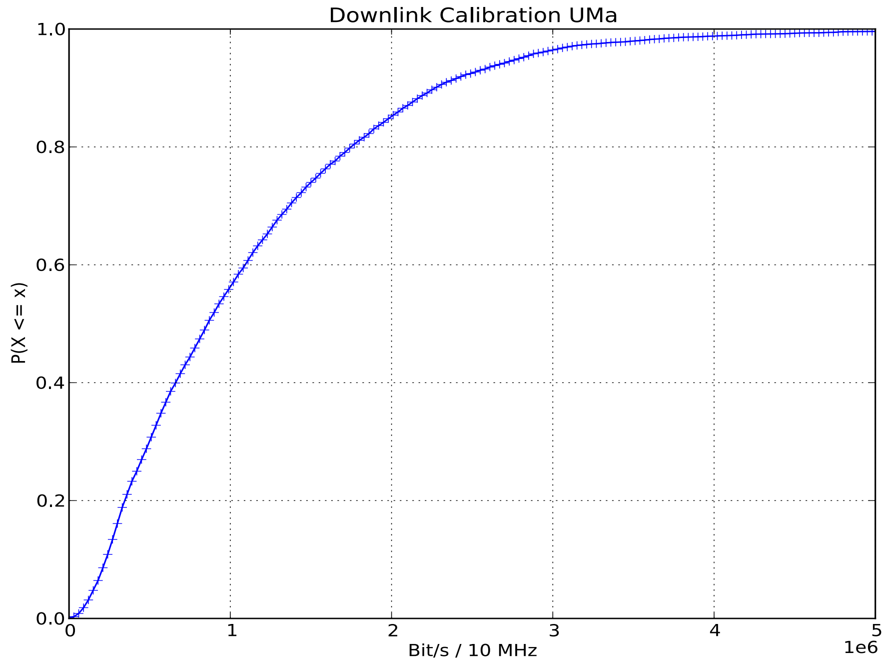




Initial System-Level Calibration Results



3GPP TR 36.814 (UMa, 1x2 MRC)



Our (current) results (UMa, 1x2 MRC)

Cell spectral efficiency: 1.0 Bit/s/Hz ± 8%

Cell edge user : 0.022 Bit/s/Hz ± 17%
(5%-tile user throughput CDF)

Cell spectral efficiency: 1.09 Bit/s/Hz

Cell edge user : 0.0154 Bit/s/Hz
(5%-tile user throughput CDF)

Simulator Benchmarking

Scenario 1: about 10s (10000 TTIs) simulation time per 1 hour wall clock time

- 57 cells, 10 users per cell, explicit interference modeling
- 10 MHz (50 PRBs)
- 1x1 antenna configuration
- Spatial channel model only on 570 serving links
- No PMI or Rank feedback computation

Scenario 2: about 1s (1000 TTIs) simulation time per 1 hour wall clock time

- 57 cells, 10 users per cell, explicit interference modeling
- 10 MHz (50 PRBs)
- 4x4 antenna configuration
- Spatial channel model on all 32490 links
- Closed-loop spatial multiplexing PMI, CQI and Rank feedback computation every 2ms for PMI/CQI on every second PRB)

Hardware: 8 cores Intel X5460 @ 3.16GHz (from 2008)