

# 5G New Radio Simulations with ns-3

**Lorenza Giupponi**

Research Director, Mobile Networks Department  
Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)

**Thomas R. Henderson**

ns-3 co-founder and lead maintainer  
Executive Director, University of Washington ns-3 Consortium

**ACMSE Conference**

**April 17, 2021**

# Tutorial goals

---

- Attendees should gain a basic understanding of the scope of the 5G New Radio module in the ns-3 discrete-event network simulator
- Attendees should be able to start running and modifying 5G New Radio example programs



# Outline

---

1. ns-3 orientation
2. Overview of NR module
3. Review of NR example program

- We will monitor the conference chat for questions
- We will take a 5 minute break at the end of each hour, and answer questions
- Slides are posted on the conference site
- Video will be posted at a later date

# Acknowledgments

---

- Tom Henderson's work on ns-3 has been supported by multiple NSF and NIST awards\*
- A team led by Lorenza Giupponi at CTTC has developed ns-3 cellular models for almost a decade
- ns-3 credits nearly 300 authors, and is built upon the ns-2 and ns-1 tools that originated in the 1990s

\*

- NSF award CNS-0551686: (2006-10)
- NSF award CNS-0958139: (2010-15)
- NSF award CNS-1836725 (2018-present)
- NSF award CNS-2016379: (2020-present)
- NIST Cooperative Agreement 70NANB17H170: (2017-2020)
- NIST Cooperative Agreement 70NANB20H179: (2020-present)

# What is 5G NR?

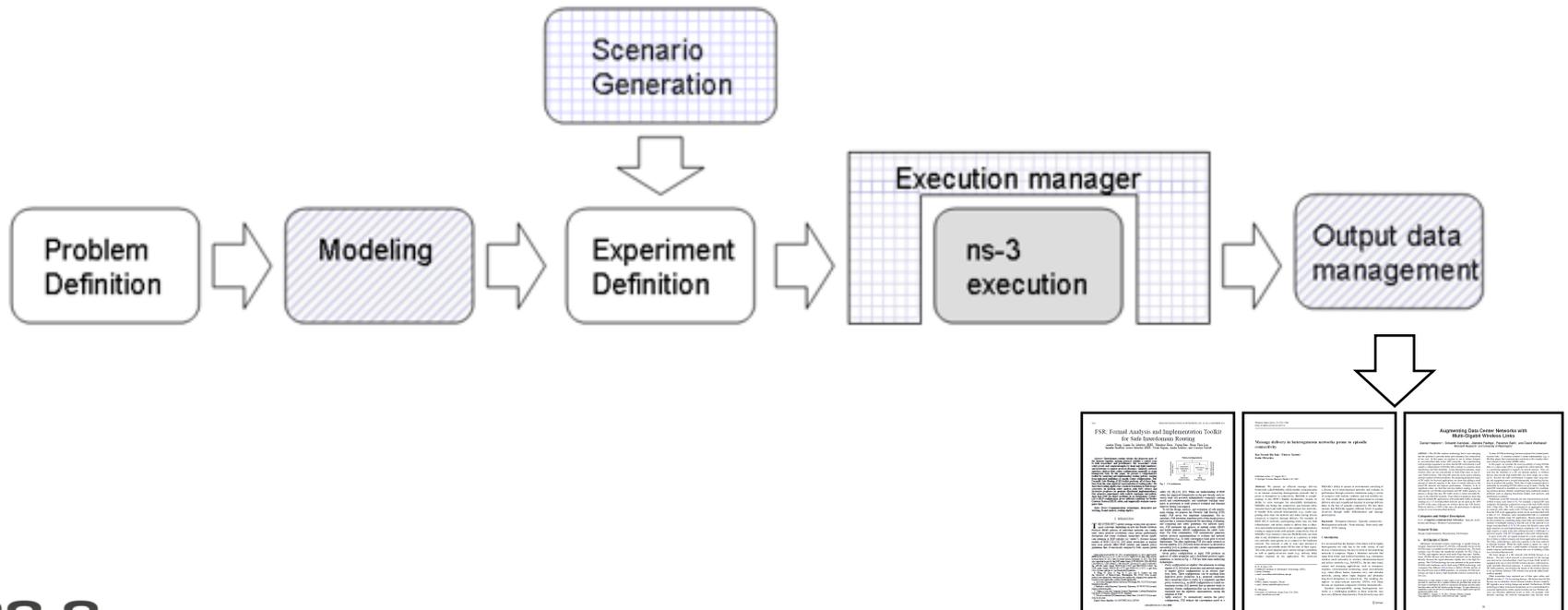
---

The next generation cellular system

- eMBB (enhanced Mobile Broadband)
- URLLC (Ultra Reliable Low Latency Communications)
- mMTC (massive Machine Type Communications)

# What is ns-3?

- A set of C++ discrete-event simulation libraries
- Community-contributed and maintained modules
- A tool to answer performance questions about computer networks



# What has ns-3 been used for?

- Thousands of publications to date — search of 'ns-3 simulator' on IEEE and ACM digital libraries, or Google Scholar

## FSR: Formal Analysis and Implementation Toolkit for Safe Interdomain Routing

Anduo Wang, Limin Jia, Member, IEEE, Wenchao Zhou, Yiyang Ren, Boon Thau Loo, Jennifer Rexford, Senior Member, IEEE, Vivek Nigam, Andre Secciani, and Carolyn Talcott

**Abstract**—Interdomain routing utilizes the disparate parts of the Internet together, making protocol stability a critical issue to both researchers and practitioners. Yet, researchers create safety proofs and counterexamples by hand and build simulators and prototypes to explore protocol dynamics. Similarly, network operators analyze their router configurations manually or using languages like Inet. In this paper, we present a comprehensive toolkit for analyzing and implementing routing policies, ranging from high-level guidelines to specific router configurations. Our *Formally Safe Routing (FSR)* toolkit performs all of these functions from the same algebraic representation of routing policy. We show that routing algebra has a natural translation to both integer constraints (to perform safety analysis with SAT solvers) and *declarative programs* (to generate distributed implementations). Our extensive experiment with realistic topologies and policies shows how *FSR* can detect problems in an autonomous system's (AS)'s BGP configurations, prove sufficient conditions for Border Gateway Protocol (BGP) safety, and empirically evaluate convergence time.

**Index Terms**—Communications technology, declarative networking, formal analysis, routing algebra.

### 1 INTRODUCTION

THE INTERNET'S global routing system does not access safety coverage, depending on how the Border Gateway Protocol (BGP) policies of individual networks are configured. Since protocol oscillations cause serious performance disruptions and some overhead, researchers derive significant attention to BGP stability (or "safety"). Abstract formal models of BGP [1]–[3], [5] allow researchers to explore how local policies affect BGP stability and identify policy guidelines that, if unilaterally adopted by ISPs, ensure global

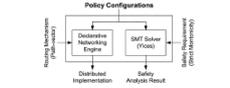


Fig. 1. FSR architecture.

to add the design, analysis, and evaluation of safe interdomain routing we propose the *Formally Safe Routing (FSR)* toolkit. *FSR* serves two important communities: For researchers, *FSR* automates important parts of the design process and provides a common framework for describing, evaluating, and comparing new safety guidelines. For network operators, *FSR* automates the analysis of internal router (BGP) and border gateway (BGP) configurations for safety violations. For both communities, *FSR* automatically generates realistic protocol implementations to evaluate real network configurations (e.g., to study convergence time) prior to actual deployment. The idea underlying *FSR* also finds resonance in routing algebra [1]–[3], [5] with recent advances in declarative networking [2] to produce provably correct implementations of safe interdomain routing.

Given policy configurations as input, *FSR* produces an analysis of safety properties and a distributed protocol implementation, as shown in Fig. 1. *FSR* has three main underlying technologies:

- **Policy configuration as algebra:** Our extension to routing algebra [1]–[3], [5] allows researchers and network operators to express policy configurations in an abstract algebraic form. These configurations can be anything from high-level policy guidelines (e.g., proposed constraints that a researcher wants to study) or a completely specified policy instance (e.g., an ISGP configuration or a multi- autonomous-system (AS) network that an operator wants to analyze). Router configuration files can be automatically translated into the algebraic representation, enabling the adoption of *FSR*.
- **Safety analysis:** To automatically analyze a policy configuration, *FSR* reduces the convergence proof to a

Manuscript received May 23, 2012; accepted January 31, 2013; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor Z. M. Mao. This work of the authors is supported in part by the NSF under Grants CCR-0620201, CNS-0804942, CNS-0804943, CNS-0804944, CNS-0804945, CNS-0804946, CNS-0804947, CNS-0804948, CNS-0804949, CNS-0804950, CNS-0804951, CNS-0804952, CNS-0804953, CNS-0804954, CNS-0804955, CNS-0804956, CNS-0804957, CNS-0804958, CNS-0804959, CNS-0804960, CNS-0804961, CNS-0804962, CNS-0804963, CNS-0804964, CNS-0804965, CNS-0804966, CNS-0804967, CNS-0804968, CNS-0804969, CNS-0804970, CNS-0804971, CNS-0804972, CNS-0804973, CNS-0804974, CNS-0804975, CNS-0804976, CNS-0804977, CNS-0804978, CNS-0804979, CNS-0804980, CNS-0804981, CNS-0804982, CNS-0804983, CNS-0804984, CNS-0804985, CNS-0804986, CNS-0804987, CNS-0804988, CNS-0804989, CNS-0804990, CNS-0804991, CNS-0804992, CNS-0804993, CNS-0804994, CNS-0804995, CNS-0804996, CNS-0804997, CNS-0804998, CNS-0804999, CNS-0805000, CNS-0805001, CNS-0805002, CNS-0805003, CNS-0805004, CNS-0805005, CNS-0805006, CNS-0805007, CNS-0805008, CNS-0805009, CNS-0805010, CNS-0805011, CNS-0805012, CNS-0805013, CNS-0805014, CNS-0805015, CNS-0805016, CNS-0805017, CNS-0805018, CNS-0805019, CNS-0805020, CNS-0805021, CNS-0805022, CNS-0805023, CNS-0805024, CNS-0805025, CNS-0805026, CNS-0805027, CNS-0805028, CNS-0805029, CNS-0805030, CNS-0805031, CNS-0805032, CNS-0805033, CNS-0805034, CNS-0805035, CNS-0805036, CNS-0805037, CNS-0805038, CNS-0805039, CNS-0805040, CNS-0805041, CNS-0805042, CNS-0805043, CNS-0805044, CNS-0805045, CNS-0805046, CNS-0805047, CNS-0805048, CNS-0805049, CNS-0805050, CNS-0805051, CNS-0805052, CNS-0805053, CNS-0805054, CNS-0805055, CNS-0805056, CNS-0805057, CNS-0805058, CNS-0805059, CNS-0805060, CNS-0805061, CNS-0805062, CNS-0805063, CNS-0805064, CNS-0805065, CNS-0805066, CNS-0805067, CNS-0805068, CNS-0805069, CNS-0805070, CNS-0805071, CNS-0805072, CNS-0805073, CNS-0805074, CNS-0805075, CNS-0805076, CNS-0805077, CNS-0805078, CNS-0805079, CNS-0805080, CNS-0805081, CNS-0805082, CNS-0805083, CNS-0805084, CNS-0805085, CNS-0805086, CNS-0805087, CNS-0805088, CNS-0805089, CNS-0805090, CNS-0805091, CNS-0805092, CNS-0805093, CNS-0805094, CNS-0805095, CNS-0805096, CNS-0805097, CNS-0805098, CNS-0805099, CNS-0805100, CNS-0805101, CNS-0805102, CNS-0805103, CNS-0805104, CNS-0805105, CNS-0805106, CNS-0805107, CNS-0805108, CNS-0805109, CNS-0805110, CNS-0805111, CNS-0805112, CNS-0805113, CNS-0805114, CNS-0805115, CNS-0805116, CNS-0805117, CNS-0805118, CNS-0805119, CNS-0805120, CNS-0805121, CNS-0805122, CNS-0805123, CNS-0805124, CNS-0805125, CNS-0805126, CNS-0805127, CNS-0805128, CNS-0805129, CNS-0805130, CNS-0805131, CNS-0805132, CNS-0805133, CNS-0805134, CNS-0805135, CNS-0805136, CNS-0805137, CNS-0805138, CNS-0805139, CNS-0805140, CNS-0805141, CNS-0805142, CNS-0805143, CNS-0805144, CNS-0805145, CNS-0805146, CNS-0805147, CNS-0805148, CNS-0805149, CNS-0805150, CNS-0805151, CNS-0805152, CNS-0805153, CNS-0805154, CNS-0805155, CNS-0805156, CNS-0805157, CNS-0805158, CNS-0805159, CNS-0805160, CNS-0805161, CNS-0805162, CNS-0805163, CNS-0805164, CNS-0805165, CNS-0805166, CNS-0805167, CNS-0805168, CNS-0805169, CNS-0805170, CNS-0805171, CNS-0805172, CNS-0805173, CNS-0805174, CNS-0805175, CNS-0805176, CNS-0805177, CNS-0805178, CNS-0805179, CNS-0805180, CNS-0805181, CNS-0805182, CNS-0805183, CNS-0805184, CNS-0805185, CNS-0805186, CNS-0805187, CNS-0805188, CNS-0805189, CNS-0805190, CNS-0805191, CNS-0805192, CNS-0805193, CNS-0805194, CNS-0805195, CNS-0805196, CNS-0805197, CNS-0805198, CNS-0805199, CNS-0805200, CNS-0805201, CNS-0805202, CNS-0805203, CNS-0805204, CNS-0805205, CNS-0805206, CNS-0805207, CNS-0805208, CNS-0805209, CNS-0805210, CNS-0805211, CNS-0805212, CNS-0805213, CNS-0805214, CNS-0805215, CNS-0805216, CNS-0805217, CNS-0805218, CNS-0805219, CNS-0805220, CNS-0805221, CNS-0805222, CNS-0805223, CNS-0805224, CNS-0805225, CNS-0805226, CNS-0805227, CNS-0805228, CNS-0805229, CNS-0805230, CNS-0805231, CNS-0805232, CNS-0805233, CNS-0805234, CNS-0805235, CNS-0805236, CNS-0805237, CNS-0805238, CNS-0805239, CNS-0805240, CNS-0805241, CNS-0805242, CNS-0805243, CNS-0805244, CNS-0805245, CNS-0805246, CNS-0805247, CNS-0805248, CNS-0805249, CNS-0805250, CNS-0805251, CNS-0805252, CNS-0805253, CNS-0805254, CNS-0805255, CNS-0805256, CNS-0805257, CNS-0805258, CNS-0805259, CNS-0805260, CNS-0805261, CNS-0805262, CNS-0805263, CNS-0805264, CNS-0805265, CNS-0805266, CNS-0805267, CNS-0805268, CNS-0805269, CNS-0805270, CNS-0805271, CNS-0805272, CNS-0805273, CNS-0805274, CNS-0805275, CNS-0805276, CNS-0805277, CNS-0805278, CNS-0805279, CNS-0805280, CNS-0805281, CNS-0805282, CNS-0805283, CNS-0805284, CNS-0805285, CNS-0805286, CNS-0805287, CNS-0805288, CNS-0805289, CNS-0805290, CNS-0805291, CNS-0805292, CNS-0805293, CNS-0805294, CNS-0805295, CNS-0805296, CNS-0805297, CNS-0805298, CNS-0805299, CNS-0805300, CNS-0805301, CNS-0805302, CNS-0805303, CNS-0805304, CNS-0805305, CNS-0805306, CNS-0805307, CNS-0805308, CNS-0805309, CNS-0805310, CNS-0805311, CNS-0805312, CNS-0805313, CNS-0805314, CNS-0805315, CNS-0805316, CNS-0805317, CNS-0805318, CNS-0805319, CNS-0805320, CNS-0805321, CNS-0805322, CNS-0805323, CNS-0805324, CNS-0805325, CNS-0805326, CNS-0805327, CNS-0805328, CNS-0805329, CNS-0805330, CNS-0805331, CNS-0805332, CNS-0805333, CNS-0805334, CNS-0805335, CNS-0805336, CNS-0805337, CNS-0805338, CNS-0805339, CNS-0805340, CNS-0805341, CNS-0805342, CNS-0805343, CNS-0805344, CNS-0805345, CNS-0805346, CNS-0805347, CNS-0805348, CNS-0805349, CNS-0805350, CNS-0805351, CNS-0805352, CNS-0805353, CNS-0805354, CNS-0805355, CNS-0805356, CNS-0805357, CNS-0805358, CNS-0805359, CNS-0805360, CNS-0805361, CNS-0805362, CNS-0805363, CNS-0805364, CNS-0805365, CNS-0805366, CNS-0805367, CNS-0805368, CNS-0805369, CNS-0805370, CNS-0805371, CNS-0805372, CNS-0805373, CNS-0805374, CNS-0805375, CNS-0805376, CNS-0805377, CNS-0805378, CNS-0805379, CNS-0805380, CNS-0805381, CNS-0805382, CNS-0805383, CNS-0805384, CNS-0805385, CNS-0805386, CNS-0805387, CNS-0805388, CNS-0805389, CNS-0805390, CNS-0805391, CNS-0805392, CNS-0805393, CNS-0805394, CNS-0805395, CNS-0805396, CNS-0805397, CNS-0805398, CNS-0805399, CNS-0805400, CNS-0805401, CNS-0805402, CNS-0805403, CNS-0805404, CNS-0805405, CNS-0805406, CNS-0805407, CNS-0805408, CNS-0805409, CNS-0805410, CNS-0805411, CNS-0805412, CNS-0805413, CNS-0805414, CNS-0805415, CNS-0805416, CNS-0805417, CNS-0805418, CNS-0805419, CNS-0805420, CNS-0805421, CNS-0805422, CNS-0805423, CNS-0805424, CNS-0805425, CNS-0805426, CNS-0805427, CNS-0805428, CNS-0805429, CNS-0805430, CNS-0805431, CNS-0805432, CNS-0805433, CNS-0805434, CNS-0805435, CNS-0805436, CNS-0805437, CNS-0805438, CNS-0805439, CNS-0805440, CNS-0805441, CNS-0805442, CNS-0805443, CNS-0805444, CNS-0805445, CNS-0805446, CNS-0805447, CNS-0805448, CNS-0805449, CNS-0805450, CNS-0805451, CNS-0805452, CNS-0805453, CNS-0805454, CNS-0805455, CNS-0805456, CNS-0805457, CNS-0805458, CNS-0805459, CNS-0805460, CNS-0805461, CNS-0805462, CNS-0805463, CNS-0805464, CNS-0805465, CNS-0805466, CNS-0805467, CNS-0805468, CNS-0805469, CNS-0805470, CNS-0805471, CNS-0805472, CNS-0805473, CNS-0805474, CNS-0805475, CNS-0805476, CNS-0805477, CNS-0805478, CNS-0805479, CNS-0805480, CNS-0805481, CNS-0805482, CNS-0805483, CNS-0805484, CNS-0805485, CNS-0805486, CNS-0805487, CNS-0805488, CNS-0805489, CNS-0805490, CNS-0805491, CNS-0805492, CNS-0805493, CNS-0805494, CNS-0805495, CNS-0805496, CNS-0805497, CNS-0805498, CNS-0805499, CNS-0805500, CNS-0805501, CNS-0805502, CNS-0805503, CNS-0805504, CNS-0805505, CNS-0805506, CNS-0805507, CNS-0805508, CNS-0805509, CNS-0805510, CNS-0805511, CNS-0805512, CNS-0805513, CNS-0805514, CNS-0805515, CNS-0805516, CNS-0805517, CNS-0805518, CNS-0805519, CNS-0805520, CNS-0805521, CNS-0805522, CNS-0805523, CNS-0805524, CNS-0805525, CNS-0805526, CNS-0805527, CNS-0805528, CNS-0805529, CNS-0805530, CNS-0805531, CNS-0805532, CNS-0805533, CNS-0805534, CNS-0805535, CNS-0805536, CNS-0805537, CNS-0805538, CNS-0805539, CNS-0805540, CNS-0805541, CNS-0805542, CNS-0805543, CNS-0805544, CNS-0805545, CNS-0805546, CNS-0805547, CNS-0805548, CNS-0805549, CNS-0805550, CNS-0805551, CNS-0805552, CNS-0805553, CNS-0805554, CNS-0805555, CNS-0805556, CNS-0805557, CNS-0805558, CNS-0805559, CNS-0805560, CNS-0805561, CNS-0805562, CNS-0805563, CNS-0805564, CNS-0805565, CNS-0805566, CNS-0805567, CNS-0805568, CNS-0805569, CNS-0805570, CNS-0805571, CNS-0805572, CNS-0805573, CNS-0805574, CNS-0805575, CNS-0805576, CNS-0805577, CNS-0805578, CNS-0805579, CNS-0805580, CNS-0805581, CNS-0805582, CNS-0805583, CNS-0805584, CNS-0805585, CNS-0805586, CNS-0805587, CNS-0805588, CNS-0805589, CNS-0805590, CNS-0805591, CNS-0805592, CNS-0805593, CNS-0805594, CNS-0805595, CNS-0805596, CNS-0805597, CNS-0805598, CNS-0805599, CNS-0805600, CNS-0805601, CNS-0805602, CNS-0805603, CNS-0805604, CNS-0805605, CNS-0805606, CNS-0805607, CNS-0805608, CNS-0805609, CNS-0805610, CNS-0805611, CNS-0805612, CNS-0805613, CNS-0805614, CNS-0805615, CNS-0805616, CNS-0805617, CNS-0805618, CNS-0805619, CNS-0805620, CNS-0805621, CNS-0805622, CNS-0805623, CNS-0805624, CNS-0805625, CNS-0805626, CNS-0805627, CNS-0805628, CNS-0805629, CNS-0805630, CNS-0805631, CNS-0805632, CNS-0805633, CNS-0805634, CNS-0805635, CNS-0805636, CNS-0805637, CNS-0805638, CNS-0805639, CNS-0805640, CNS-0805641, CNS-0805642, CNS-0805643, CNS-0805644, CNS-0805645, CNS-0805646, CNS-0805647, CNS-0805648, CNS-0805649, CNS-0805650, CNS-0805651, CNS-0805652, CNS-0805653, CNS-0805654, CNS-0805655, CNS-0805656, CNS-0805657, CNS-0805658, CNS-0805659, CNS-0805660, CNS-0805661, CNS-0805662, CNS-0805663, CNS-0805664, CNS-0805665, CNS-0805666, CNS-0805667, CNS-0805668, CNS-0805669, CNS-0805670, CNS-0805671, CNS-0805672, CNS-0805673, CNS-0805674, CNS-0805675, CNS-0805676, CNS-0805677, CNS-0805678, CNS-0805679, CNS-0805680, CNS-0805681, CNS-0805682, CNS-0805683, CNS-0805684, CNS-0805685, CNS-0805686, CNS-0805687, CNS-0805688, CNS-0805689, CNS-0805690, CNS-0805691, CNS-0805692, CNS-0805693, CNS-0805694, CNS-0805695, CNS-0805696, CNS-0805697, CNS-0805698, CNS-0805699, CNS-0805700, CNS-0805701, CNS-0805702, CNS-0805703, CNS-0805704, CNS-0805705, CNS-0805706, CNS-0805707, CNS-0805708, CNS-0805709, CNS-0805710, CNS-0805711, CNS-0805712, CNS-0805713, CNS-0805714, CNS-0805715, CNS-0805716, CNS-0805717, CNS-0805718, CNS-0805719, CNS-0805720, CNS-0805721, CNS-0805722, CNS-0805723, CNS-0805724, CNS-0805725, CNS-0805726, CNS-0805727, CNS-0805728, CNS-0805729, CNS-0805730, CNS-0805731, CNS-0805732, CNS-0805733, CNS-0805734, CNS-0805735, CNS-0805736, CNS-0805737, CNS-0805738, CNS-0805739, CNS-0805740, CNS-0805741, CNS-0805742, CNS-0805743, CNS-0805744, CNS-0805745, CNS-0805746, CNS-0805747, CNS-0805748, CNS-0805749, CNS-0805750, CNS-0805751, CNS-0805752, CNS-0805753, CNS-0805754, CNS-0805755, CNS-0805756, CNS-0805757, CNS-0805758, CNS-0805759, CNS-0805760, CNS-0805761, CNS-0805762, CNS-0805763, CNS-0805764, CNS-0805765, CNS-0805766, CNS-0805767, CNS-0805768, CNS-0805769, CNS-0805770, CNS-0805771, CNS-0805772, CNS-0805773, CNS-0805774, CNS-0805775, CNS-0805776, CNS-0805777, CNS-0805778, CNS-0805779, CNS-0805780, CNS-0805781, CNS-0805782, CNS-0805783, CNS-0805784, CNS-0805785, CNS-0805786, CNS-0805787, CNS-0805788, CNS-0805789, CNS-0805790, CNS-0805791, CNS-0805792, CNS-0805793, CNS-0805794, CNS-0805795, CNS-0805796, CNS-0805797, CNS-0805798, CNS-0805799, CNS-0805800, CNS-0805801, CNS-0805802, CNS-0805803, CNS-0805804, CNS-0805805, CNS-0805806, CNS-0805807, CNS-0805808, CNS-0805809, CNS-0805810, CNS-0805811, CNS-0805812, CNS-0805813, CNS-0805814, CNS-0805815, CNS-0805816, CNS-0805817, CNS-0805818, CNS-0805819, CNS-0805820, CNS-0805821, CNS-0805822, CNS-0805823, CNS-0805824, CNS-0805825, CNS-0805826, CNS-0805827, CNS-0805828, CNS-0805829, CNS-0805830, CNS-0805831, CNS-0805832, CNS-0805833, CNS-0805834, CNS-0805835, CNS-0805836, CNS-0805837, CNS-0805838, CNS-0805839, CNS-0805840, CNS-0805841, CNS-0805842, CNS-0805843, CNS-0805844, CNS-0805845, CNS-0805846, CNS-0805847, CNS-0805848, CNS-0805849, CNS-0805850, CNS-0805851, CNS-0805852, CNS-0805853, CNS-0805854, CNS-0805855, CNS-0805856, CNS-0805857, CNS-0805858, CNS-0805859, CNS-0805860, CNS-0805861, CNS-0805862, CNS-0805863, CNS-0805864, CNS-0805865, CNS-0805866, CNS-0805867, CNS-0805868, CNS-0805869, CNS-0805870, CNS-0805871, CNS-0805872, CNS-0805873, CNS-0805874, CNS-0805875, CNS-0805876, CNS-0805877, CNS-0805878, CNS-0805879, CNS-0805880, CNS-0805881, CNS-0805882, CNS-0805883, CNS-0805884, CNS-0805885, CNS-0805886, CNS-0805887, CNS-0805888, CNS-0805889, CNS-0805890, CNS-0805891, CNS-0805892, CNS-0805893, CNS-0805894, CNS-0805895, CNS-0805896, CNS-0805897, CNS-0805898, CNS-0805899, CNS-0805900, CNS-0805901, CNS-0805902, CNS-0805903, CNS-0805904, CNS-0805905, CNS-0805906, CNS-0805907, CNS-0805908, CNS-0805909, CNS-0805910, CNS-0805911, CNS-0805912, CNS-0805913, CNS-0805914, CNS-0805915, CNS-0805916, CNS-0805917, CNS-0805918, CNS-0805919, CNS-0805920, CNS-0805921, CNS-0805922, CNS-0805923, CNS-0805924, CNS-0805925, CNS-0805926, CNS-0805927, CNS-0805928, CNS-0805929, CNS-0805930, CNS-0805931, CNS-0805932, CNS-0805933, CNS-0805934, CNS-0805935, CNS-0805936, CNS-0805937, CNS-0805938, CNS-0805939, CNS-0805940, CNS-0805941, CNS-0805942, CNS-0805943, CNS-0805944, CNS-0805945, CNS-0805946, CNS-0805947, CNS-0805948, CNS-0805949, CNS-0805950, CNS-0805951, CNS-0805952, CNS-0805953, CNS-0805954, CNS-0805955, CNS-0805956, CNS-0805957, CNS-0805958, CNS-0805959, CNS-0805960, CNS-0805961, CNS-0805962, CNS-0805963, CNS-0805964, CNS-0805965, CNS-0805966, CNS-0805967, CNS-0805968, CNS-0805969, CNS-0805970, CNS-0805971, CNS-0805972, CNS-0805973, CNS-0805974, CNS-0805975, CNS-0805976, CNS-0805977, CNS-0805978, CNS-0805979, CNS-0805980, CNS-0805981, CNS-0805982, CNS-0805983, CNS-0805984, CNS-0805985, CNS-0805986, CNS-0805987, CNS-0805988, CNS-0805989, CNS-0805990, CNS-0805991, CNS-0805992, CNS-0805993, CNS-0805994, CNS-0805995, CNS-0805996, CNS-0805997, CNS-0805998, CNS-0805999, CNS-0806000, CNS-0806001, CNS-0806002, CNS-0806003, CNS-0806004, CNS-0806005, CNS-0806006, CNS-0806007, CNS-0806008, CNS-0806009, CNS-0806010, CNS-0806011, CNS-0806012, CNS-0806013, CNS-0806014, CNS-0806015, CNS-0806016, CNS-0806017, CNS-0

# Why simulation for wireless networks?

---

Simulation is commonly used for wireless research, for the following reasons:

- 1) experiments can be **replicated exactly**
- 2) testbeds often do not provide **low-level access** to change how devices work
- 3) **tests at scale are hard** or expensive to orchestrate
- 4) there may be **frequency coordination**/interference concerns

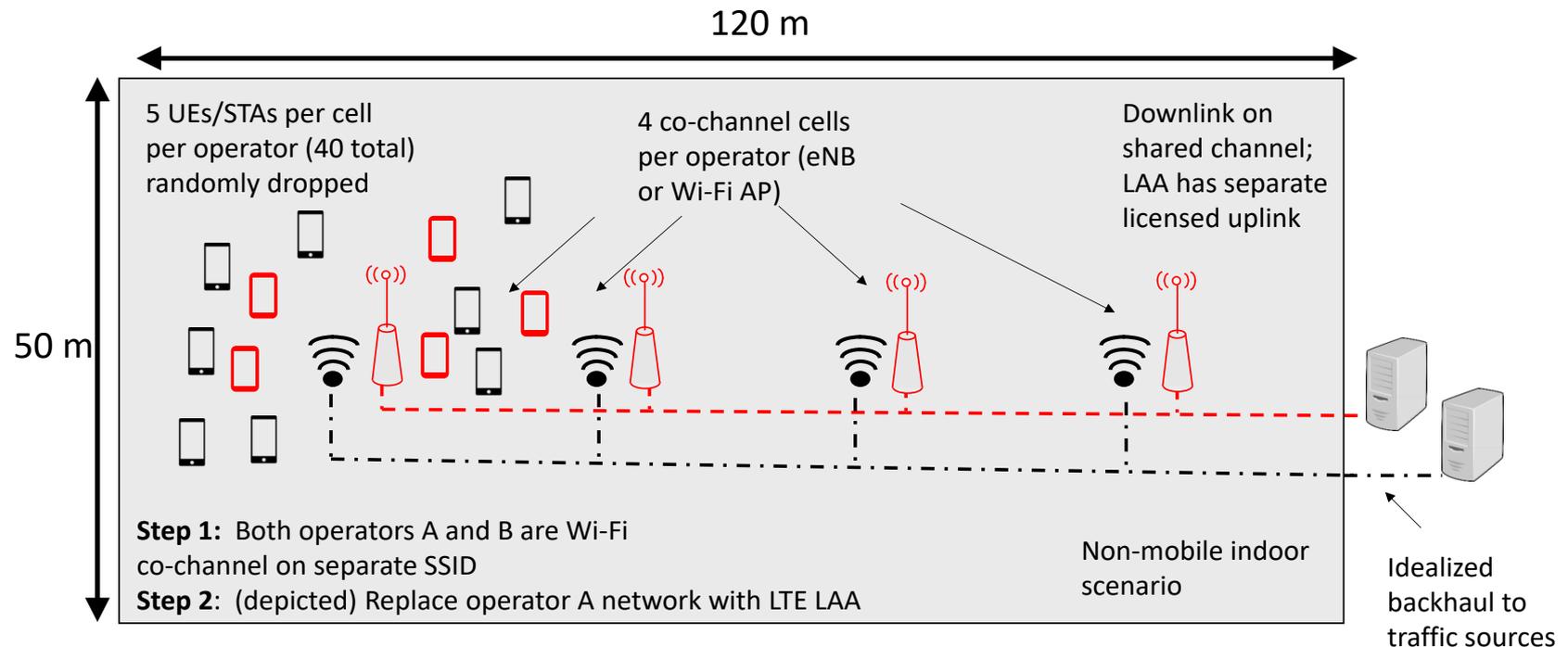
Simulation is also used in the technical standardization process; e.g.

- IEEE 802.11-14/0571r12, 11ax Evaluation Methodology
- 3GPP TR 36.889: Study on Licensed-Assisted Access to Unlicensed Spectrum

# Example performance study

## LTE Licensed-Assisted Access (LAA) coexistence

- "Can a specific unlicensed variant of LTE (LAA) operate in the same spectrum as a Wi-Fi network, without impacting Wi-Fi system throughput and latency more than another co-located Wi-Fi network would impact it?"



# Configuration details

Unlicensed channel model	3GPP TR 36.889	ns-3 implementation
Network Layout	Indoor scenario	Indoor scenario
System bandwidth	20 MHz	20 MHz
Carrier frequency	5 GHz	5 GHz (channel 36, tunable)
Number of carriers	1, 4 (to be shared between two operators) 1 for evaluations with DL+UL Wi-Fi coexisting with DL-only LAA	1 for evaluations with DL+UL Wi-Fi coexisting with DL-only LAA
Total Base Station (BS) transmission power	18/24 dBm	18/24 dBm Simulations herein consider 18 dBm
Total User equipment (UE) transmission power	18 dBm for unlicensed spectrum	18 dBm
Distance dependent path loss, shadowing and fading	ITU InH	802.11ax indoor model
Antenna pattern	2D Omni-directional	2D Omni-directional
Antenna height	6 m	6 m (LAA, not modelled for Wi-Fi)
UE antenna height	1.5 m	1.5 m (LAA, not modelled for Wi-Fi)
Antenna gain	5 dBi	5 dBi
UE antenna gain	0 dBi	0 dBi
Number of UEs	10 UEs per unlicensed band carrier per operator for DL-only 10 UEs per unlicensed band carrier per operator for DL-only for four unlicensed carriers. 20 UEs per unlicensed band carrier per operator for DL+UL for single unlicensed carrier. 20 UEs per unlicensed band carrier per operator for DL+UL Wi-Fi coexisting with DL-only LAA	Supports all the configurations in TR 36.889. Simulations herein consider the case of 20 UEs per unlicensed band carrier per operator for DL LAA coexistence evaluations for single unlicensed carrier.
UE Dropping	All UEs should be randomly dropped and be within coverage of the small cell in the unlicensed band.	Randomly dropped and within small cell coverage.
Traffic Model	FTP Model 1 and 3 based on TR 36.814 FTP model file size: 0.5 Mbytes. Optional: VoIP model based on TR36.889	FTP Model 1 as in TR36.814. FTP model file size: 0.5 Mbytes Voice model: DL only
UE noise figure	9 dB	9 dB
Cell selection	For LAA UEs, cell selection is based on RSRP (Reference Signal Received Power). For Wi-Fi stations (STAs), cell selection is based on RSS (Received signal power strength) of WiFi Access Points (APs). RSS threshold is -82 dBm.	RSRP for LAA UEs and RSS for Wi-Fi STAs
Network synchronization	For the same operator, the network can be synchronized. Small cells of different operators are not synchronized.	Small cells are synchronized, different operators are not synchronized.

Figure from: [Giupponi et al., Simulating LTE and Wi-Fi Coexistence in Unlicensed Spectrum with ns-3](#)

## 3GPP TR 36.889 V13.0.0 (2015-06)

Technical Report

**3rd Generation Partnership Project;  
Technical Specification Group Radio Access Network;  
Study on Licensed-Assisted Access to Unlicensed Spectrum;  
(Release 13)**

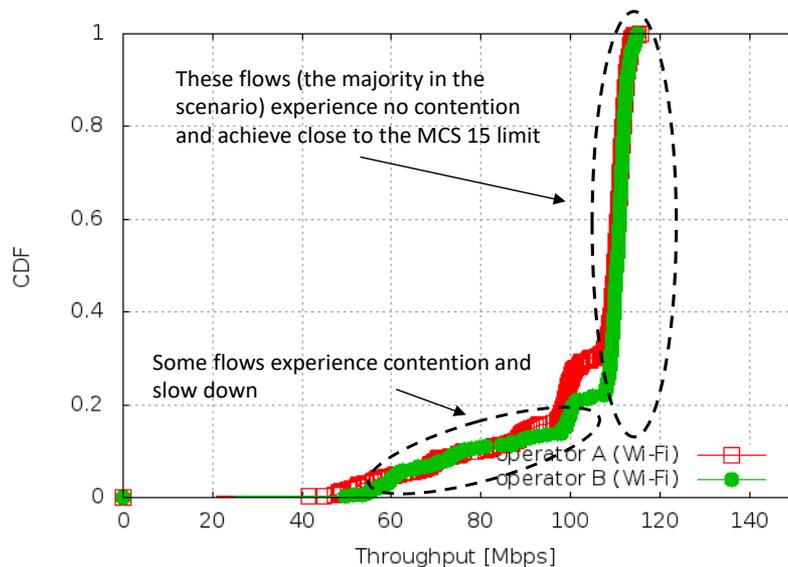



The present document has been developed within the 3rd Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPP Organizational Partners and shall not be implemented. This Report is provided for future development work within 3GPP only. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP™ systems should be obtained via the 3GPP Organizational Partners Publications Offices.

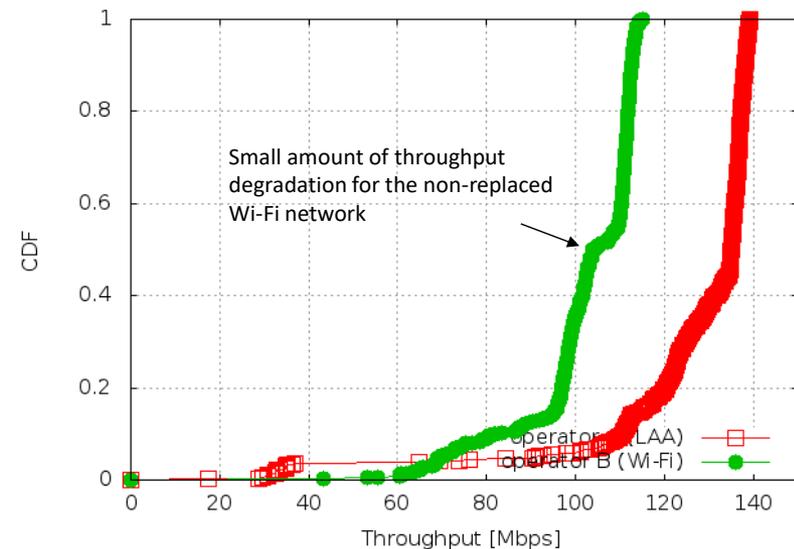
Configuration drawn from TR 36.889

# Sample results

- Place two Wi-Fi networks in same region, fully load the system, and plot a CDF of observed throughputs per station. Repeat by replacing one Wi-Fi network with LAA.



a) Step 1 (Wi-Fi)



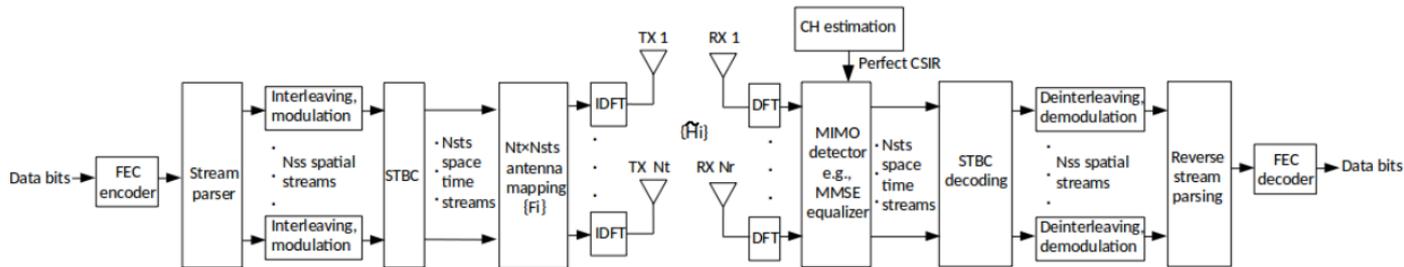
b) Step 2 (LAA)

Figures from ns-3 LAA Wi-Fi Coexistence simulation project

This is exemplary of an ns-3 performance study

# Simulation alternatives

- “Link” or physical-layer simulations
  - [MATLAB](#), [Vienna LTE-A Simulator](#)
  - Unit of granularity is the symbol (high-level of PHY realism)



Sian Jin, Sumit Roy, Weihua Jiang, and Thomas R. Henderson. 2020. Efficient Abstractions for Implementing TGN Channel and OFDM-MIMO Links in ns-3. In Proceedings of the 2020 Workshop on ns-3 (WNS3 2020).

modified from T. Paul and T. Ogunfrunmiri. 2008. Wireless LAN Comes of Age: Understanding the IEEE 802.11n Amendment. IEEE Circuits and Systems Magazine 8, 1 (First 2008), 28–54.

$N_t \times N_r (= N_{SS})$	Bandwidth	MATLAB Full-link
1 × 1	20MHz	28 min
1 × 1	40MHz	25 min
2 × 2	20MHz	37 min
2 × 2	40MHz	39 min
3 × 3	20MHz	51 min
3 × 3	40MHz	60 min

Run-times for  
40,000 simulated  
packets between  
two nodes

Table 2 from Jin et al (cited above)

# Simulation alternatives (cont.)

- “Packet-level” or full-stack simulations
  - The main unit of granularity is the packet
  - ns-2, ns-3, OMNeT++, OPNET, QualNet

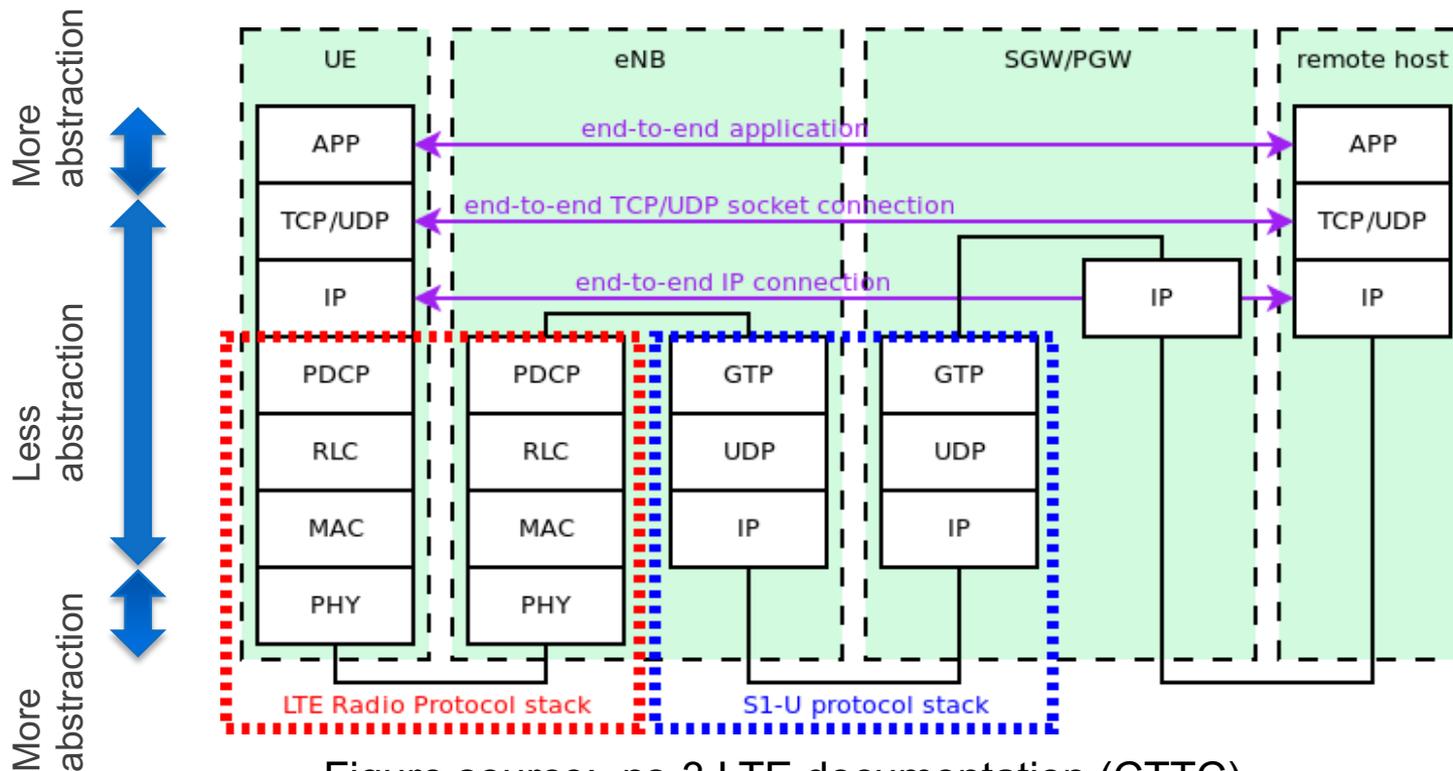


Figure source: ns-3 LTE documentation (CTTC)

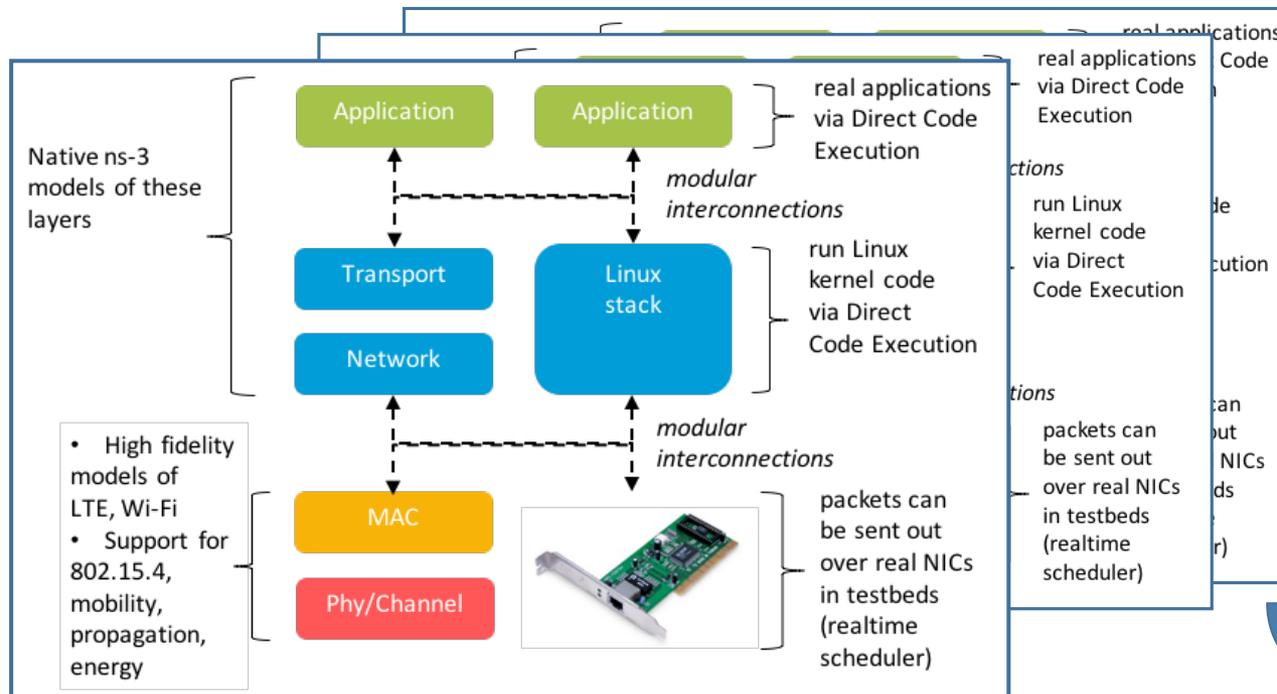
# Simulation alternatives (cont.)

---

- “System-level” or flow simulations
  - Individual packets are not modeled, but the effects of messages exchanged or packet aggregates (flows)
  - examples: [ExtendSim](#), [SimGrid](#)

# ns-3 architecture

- ns-3 is a leading **open source, packet-level network simulator** oriented towards network research, featuring a **high-performance core** enabling **parallelization across a cluster** (for large scenarios), **ability to run real code**, and **interaction with testbeds**



Runs on a single machine



or partitioned across a cluster

# Outline

---

1. ns-3 orientation
  - Obtaining the code
  - Configuring and building ns-3
  - Walkthrough of simple example: `udp-client-server.cc`
2. Overview of NR module
  - Assumptions and Architecture
  - PHY
  - MAC
  - Validation
  - Extensions (NR-U, NR-V2X)
3. Review of existing NR example
  - `cttc-nr-demo.cc`
  - pointers to documentation of other examples

# Software orientation

---

ns-3 is written in C++, with Python bindings

- Users need to work at the Linux (or Unix, macOS, etc.) command-line, compile the code, and handle raw output data

Key differences from other network simulators:

1) Command-line, Unix orientation

- vs. Integrated Development Environment (IDE)

2) Simulations and models written directly in C++ and Python

- vs. a domain-specific simulation language

# ns-3 not written in a high-level language

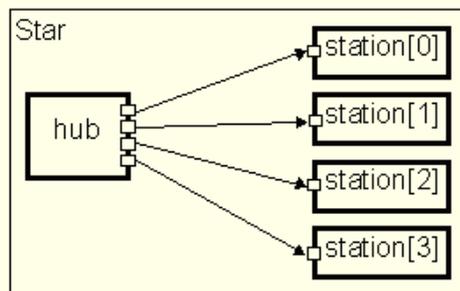
Submodule vectors, gate vectors and multiple connections are illustrated in the following example:

```
simple Hub
  gates:
    out: output[];
endsimple

simple Station //...

module Star
  submodules:
    hub: Hub
    gatesizes: output[4];
    station: Station[4];
  connections:
    for i=0..3 do
      hub.output[i] --> station[i].in;
    endfor
endmodule
```

The result of the above is depicted in Fig.4.



Example of OMNeT++ Network Description (NED) language

Figure excerpted from <http://www.ewh.ieee.org/soc/es/Nov1999/18/ned.htm>

# ns-3 does not have a graphical IDE

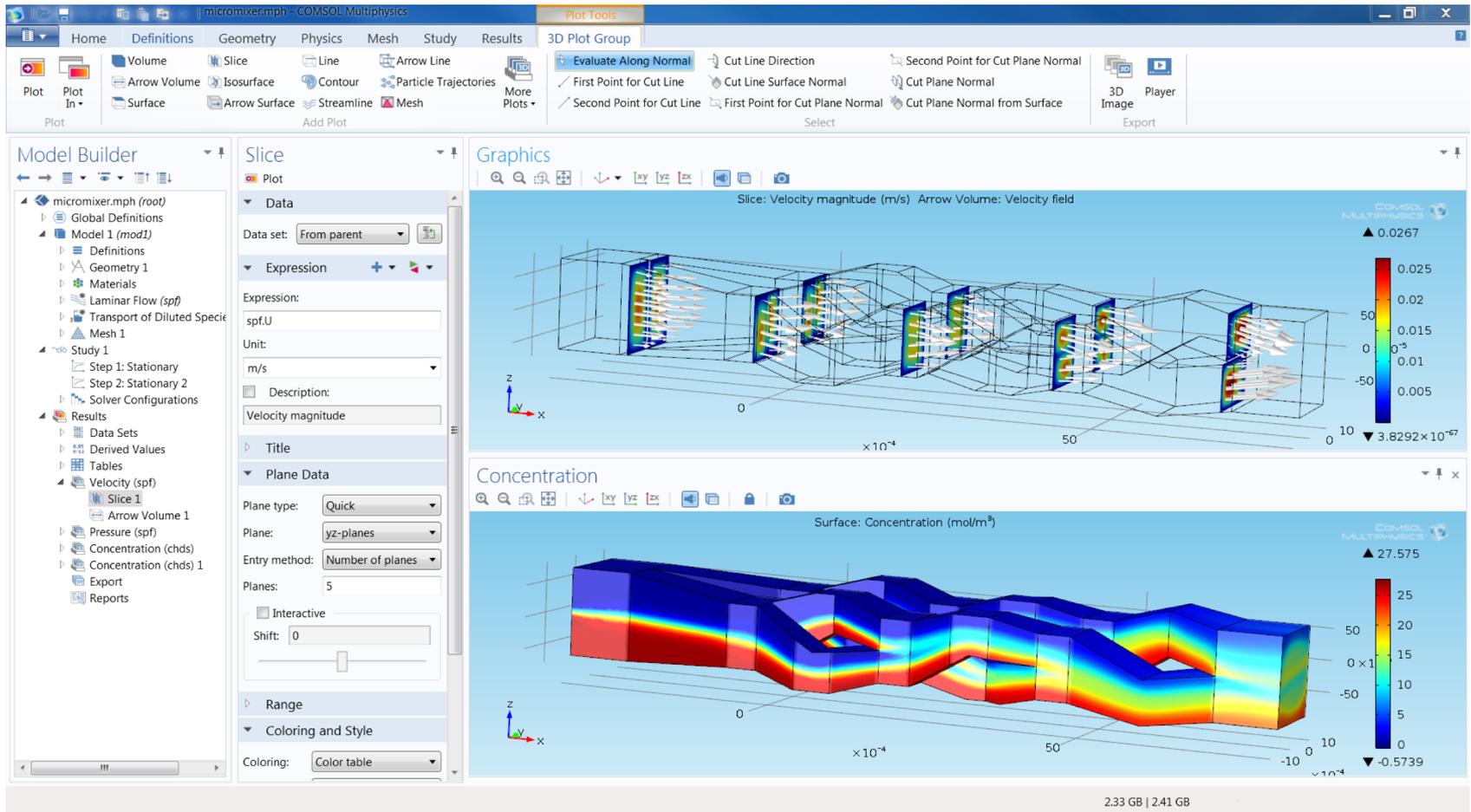
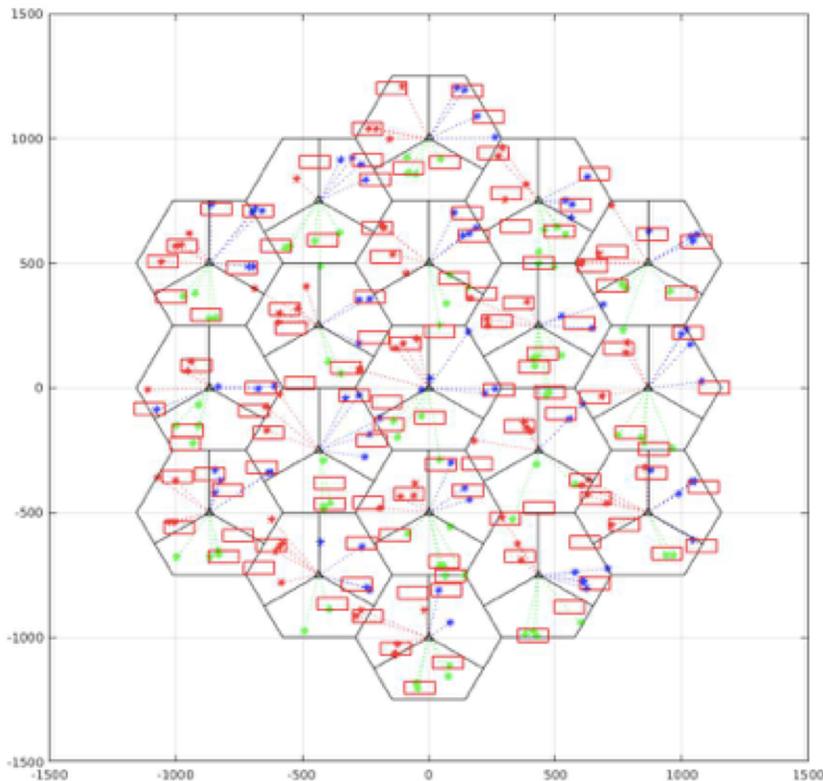


Figure source: <https://www.comsol.com/comsol-multiphysics>

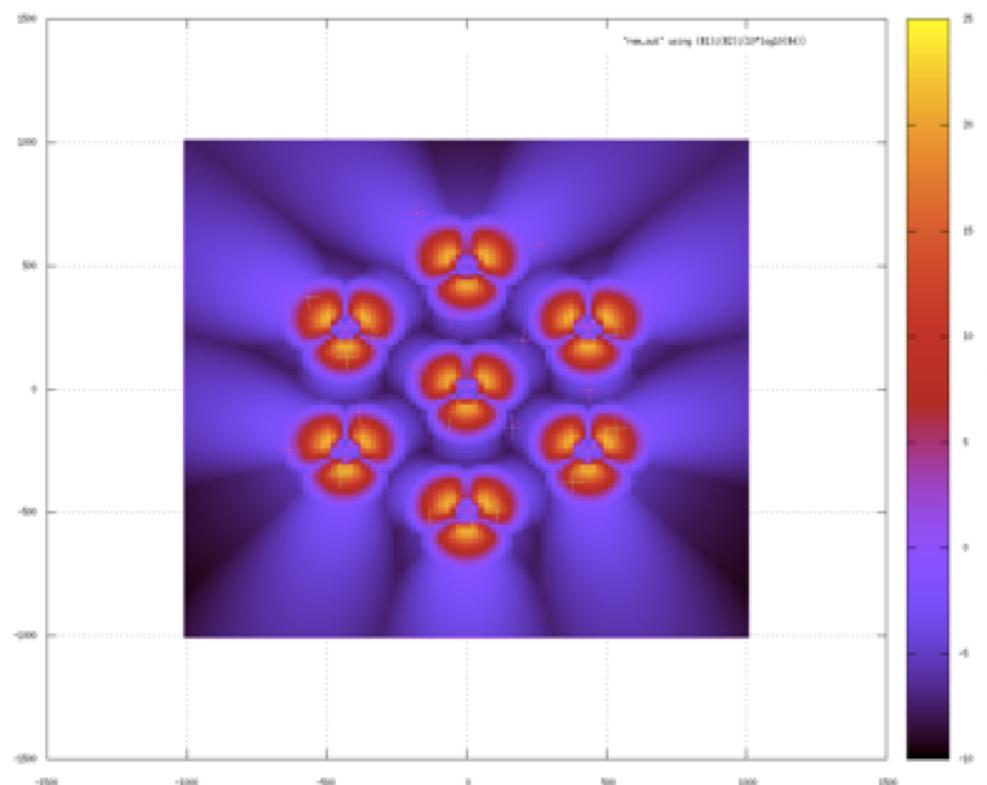
# ns-3 uses external programs for graphics

Network cell structure showing eNBs, UEs, and buildings



Matplotlib

Radio environment map showing signal strength from eNBs



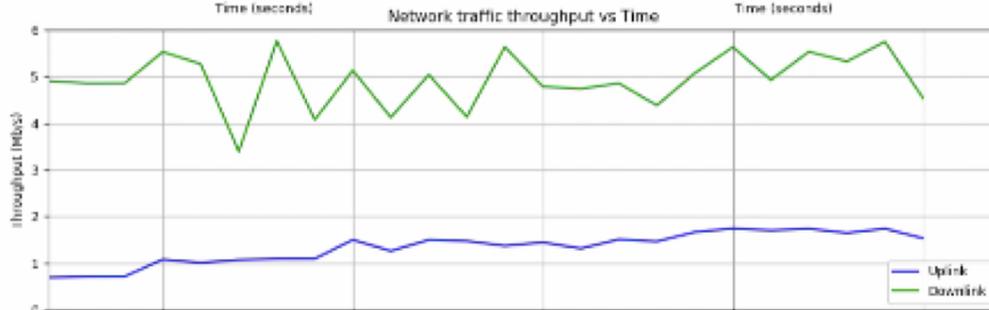
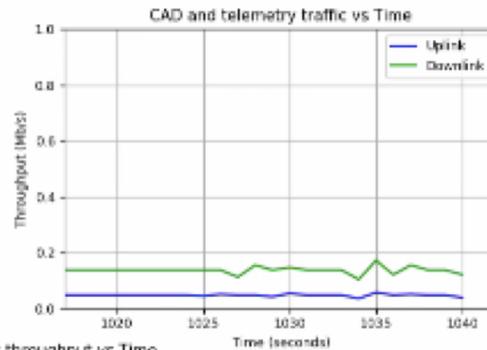
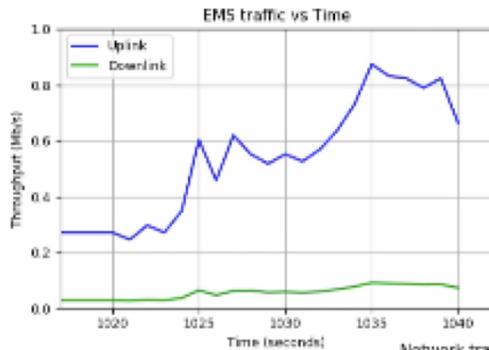
gnuplot

# ns-3 users write scripts for plotting

Plots or animations

EmsVideo\_1\_Server 942.36392953 RX 1012 1061 U 2395  
 EmsVideo\_1\_Client 942.37317727 TX 1012 1061 U 2398  
 EmsVideo\_1\_Client 942.377 RX 64 113 U 2397  
 WebBrowsingGraphics\_0\_Server 942.38092876 TX 1024 1073 U 2399  
 WebBrowsingGraphics\_0\_Client 942.394 RX 1024 1073 U 2399  
 AvlAssetPerimeter\_1\_Server 942.42492988 RX 1408 1457 U 2401

Throughput vs. time for incident scenario



Used to measure KPIs

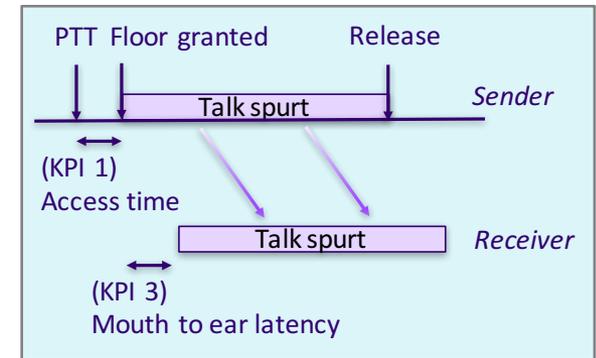
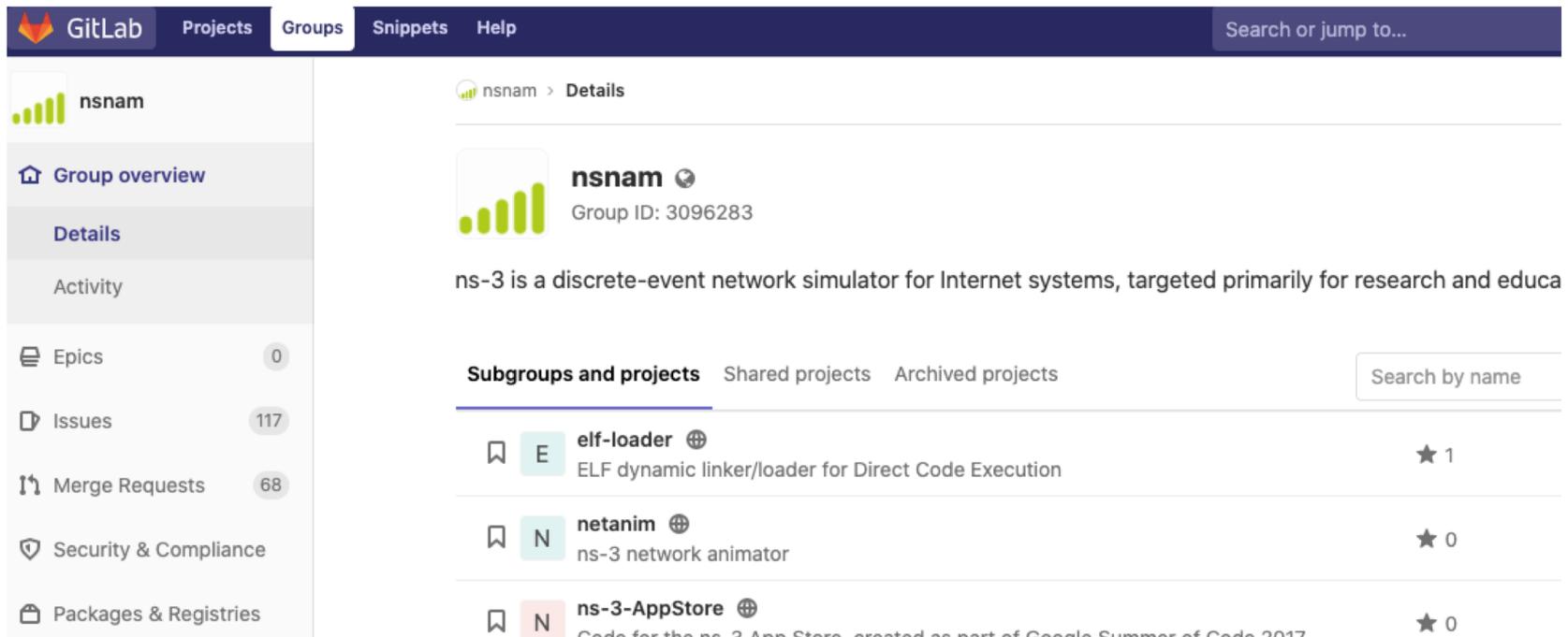


Figure source: 3GPP TS 22.179

# ns-3 software repositories

- The main web site (documentation, wiki) is <https://www.nsnam.org>
- Software is hosted in Git repositories at <https://gitlab.com/nsnam/>



The screenshot shows the GitLab web interface for the 'nsnam' group. The navigation bar at the top includes 'GitLab', 'Projects', 'Groups', 'Snippets', and 'Help', along with a search bar. The left sidebar shows the 'nsnam' group overview with options for 'Group overview', 'Details', 'Activity', 'Epics' (0), 'Issues' (117), 'Merge Requests' (68), 'Security & Compliance', and 'Packages & Registries'. The main content area displays the 'nsnam' group details, including the group name, ID (3096283), and a description: 'ns-3 is a discrete-event network simulator for Internet systems, targeted primarily for research and educa'. Below this, there is a section for 'Subgroups and projects' with a search bar and a list of projects:

Project Name	Description	Stars
elf-loader	ELF dynamic linker/loader for Direct Code Execution	★ 1
netanim	ns-3 network animator	★ 0
ns-3-AppStore	Code for the ns-3 App Store, created as part of Google Summer of Code 2017	★ 0

# ns-3 App Store

- <https://apps.nsnam.org>



**All Apps**

**Categories**  

- Routing
- MANET
- Underwater Acoustic Networks
- LTE
- Public Safety
- Reinforcement Learning
- Satellite Networking
- Transport Protocols
- Utilities
- Named Data Networking

## Newest Releases

[Get Started with the App Store »](#)



**if77 model for ns-3**

Wrapper around the if77 propagation model.



**EMPIRE Shim**

Shim to enable use of the EMPIRE propagation suite in ns-3



**DOCSIS models for ns-3**

This module extends ns-3 to simulate the MAC layer operation

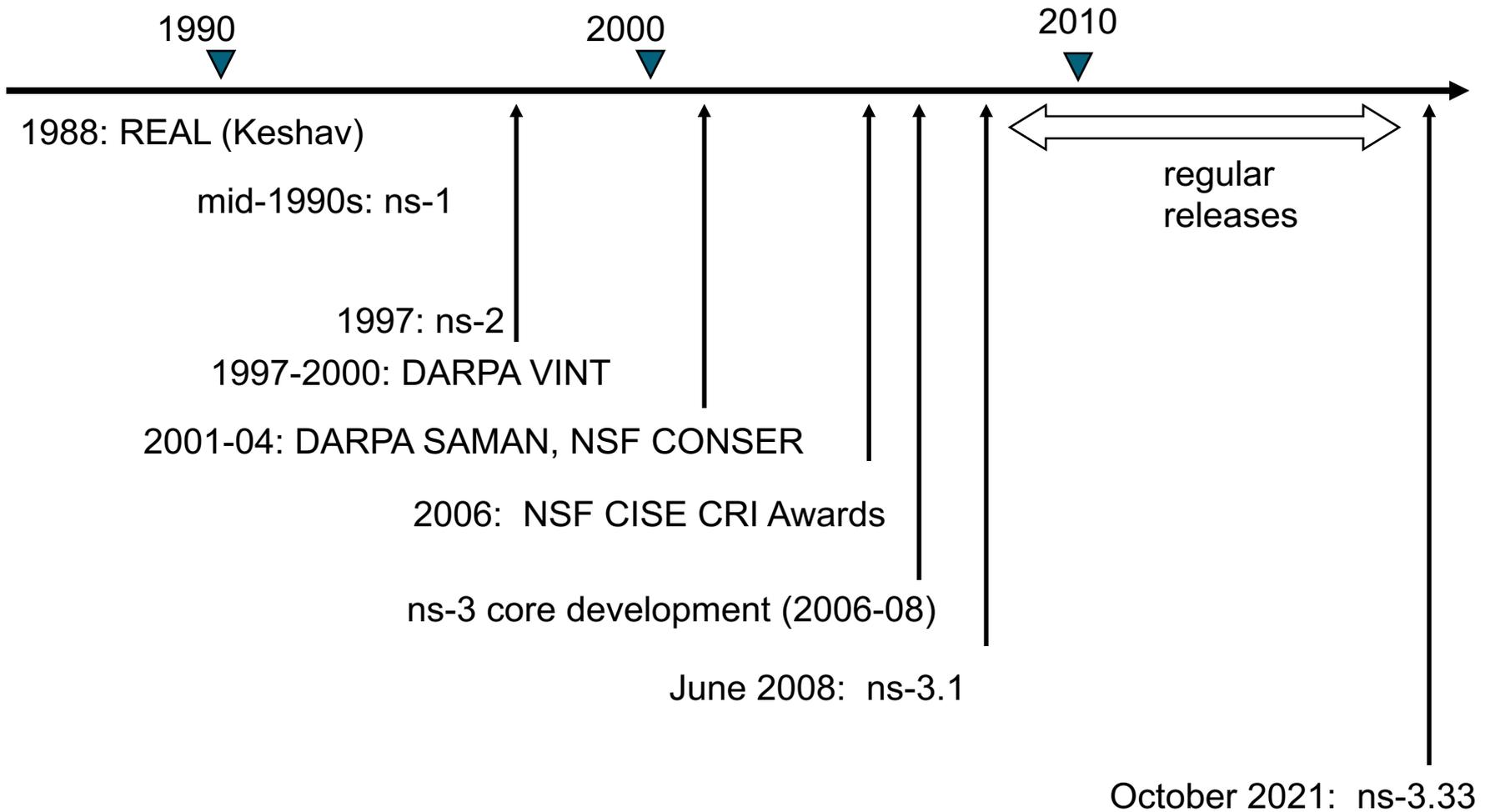


**Direct Code Execution**

Direct Code Execution (DCE) is a framework to run real

[more newest releases »](#)

# ns timeline



# An open source project

---

Between mid-2019 and mid-2020:

- **578 commits** by **51 authors** (33 new!)
- **Maintainer commits** from
  - Alexander Krotov, Biljana Bojovic, Manuel Requena, Mohit Tahiliani, Natale Patriciello, Peter Barnes, Getachew Redietab, Sebastien Deronne, Stefano Avallone, Tom Henderson, Tommaso Pecorella, Zoraze Ali
- **72,696 lines** of C++ code added/deleted
  - parsed output of `git diff --stat`
- **261 Merge Requests** opened
- **160 Issues** filed

# Links to videos, tutorials

---

- To get started using ns-3
  - [ns-3 tutorial](#)
  - [ns-3 training videos](#) (2019 annual meeting)
    - special sessions for TCP, Wi-Fi, LTE, sensor networks
  - [Other videos](#) (available also by searching YouTube)
- More information about ns-3 and wireless
  - [2019 Workshop on Next-Generation Wireless with ns-3 \(WNGW\)](#)
    - many talks on a variety of wireless technologies, from industry and academia

# 4G LTE models in ns-3

---

- LENA: An open source product-oriented LTE/EPC Network Simulator
  - Developed by CTTC, Barcelona from 2012-present
  - Designed around an industrial API: the Small Cell Forum MAC Scheduler Interface Specification
  - Full stack, end-to-end
  - Accurate model of the LTE/EPC protocol stack
  - Specific Channel and PHY layer models for LTE macro and small cells
  - Emphasis on radio-level performance and end-to-end QoE
  - Extended for public safety networks (NIST, UW, CTTC, in the ns-3 App Store)

# 5G NR module (this tutorial)

---

- 5G NR (led by [CTTC, Barcelona](#))
  - Leverages [mmWave models](#) developed by the University of Padova and NYU Wireless
  - Supports fundamental PHY-MAC NR features aligned with NR Release 15 TS 38.300
  - Additional recent work to support V2X and NR Unlicensed



# Outline

---

1. ns-3 orientation
  - Obtaining the code
  - Configuring and building ns-3
  - Walkthrough of simple example: `udp-client-server.cc`
2. Overview of NR module
  - Assumptions and Architecture
  - PHY
  - MAC
  - Validation
  - Extensions (NR-U, NR-V2X)
3. Review of existing NR example
  - `cttc-nr-demo.cc`
  - pointers to documentation of other examples

# Obtaining the code

---

- For working with NR, we recommend using the development version of ns-3 along with the latest release of 5G LENA
  - ns-3-dev: git clone <https://gitlab.com/nsnam/ns-3-dev.git>
  - 5G NR: git clone <https://gitlab.com/cttc-lena/nr.git>
- Caveats:
  1. access to 5G NR must be requested from CTTC
  2. the 5G NR module must be downloaded separately as a contributed module
  3. we only support recent versions of Linux or macOS Command-Line Tools

# Why a separately downloaded module?

---

- Useful to track impact
  - Supporting a module such as nr requires significant amount of institutional resources
  - CTTC benefits from this arrangement to better track the dissemination and impact of open sourcing this module for ns-3
- ns-3 in general is moving to a more federated development process through its App Store

# Obtaining access to 5G NR module

- Visit <https://5g-lena.cttc.es>



Features Publications Docs Services Download Blog Contact

## 5G-LENA simulator

### *ns-3 module to simulate 3GPP 5G networks*

5G-LENA is a GPLv2 New Radio (NR) network simulator, designed as a pluggable module to [ns-3](#). Its development, initially funded by InterDigital, is open to the community in order to foster early adoption, contributions by industrial and academic partners, collaborative development and results reproducibility.

The simulator is the natural evolution of [LENA](#), the LTE/EPC Network Simulator, the development started from the [mmWave module](#), and it incorporates fundamental PHY-MAC NR features aligned with NR Release 15 TS 38.300. The first release of the simulator is dated February 2019.

### Get the software

along with complete  
source code

### Join 5G-LENA Group

to discuss, collaborate  
and share results

# Quick start

---

```
$ git clone https://gitlab.com/nsnam/ns-3-dev.git
$ cd ns-3-dev/contrib
$ git clone https://gitlab.com/cttc-lena/nr.git
$ cd nr
$ $ git checkout -b 5g-lena-v1.1.y origin/5g-lena-
v1.1.y
```

Branch 5g-lena-v1.1.y set up to track remote branch 5g-lena-v1.1.y from origin.

Switched to a new branch '5g-lena-v1.1.y'

```
$ cd ../../
$ ./waf configure -d optimized --enable-examples --
enable-tests --enable-logs
$ ./waf build
$ ./test.py
$ ./waf --run cttc-nr-demo
```

# Documentation

---

Extensive documentation at both sites

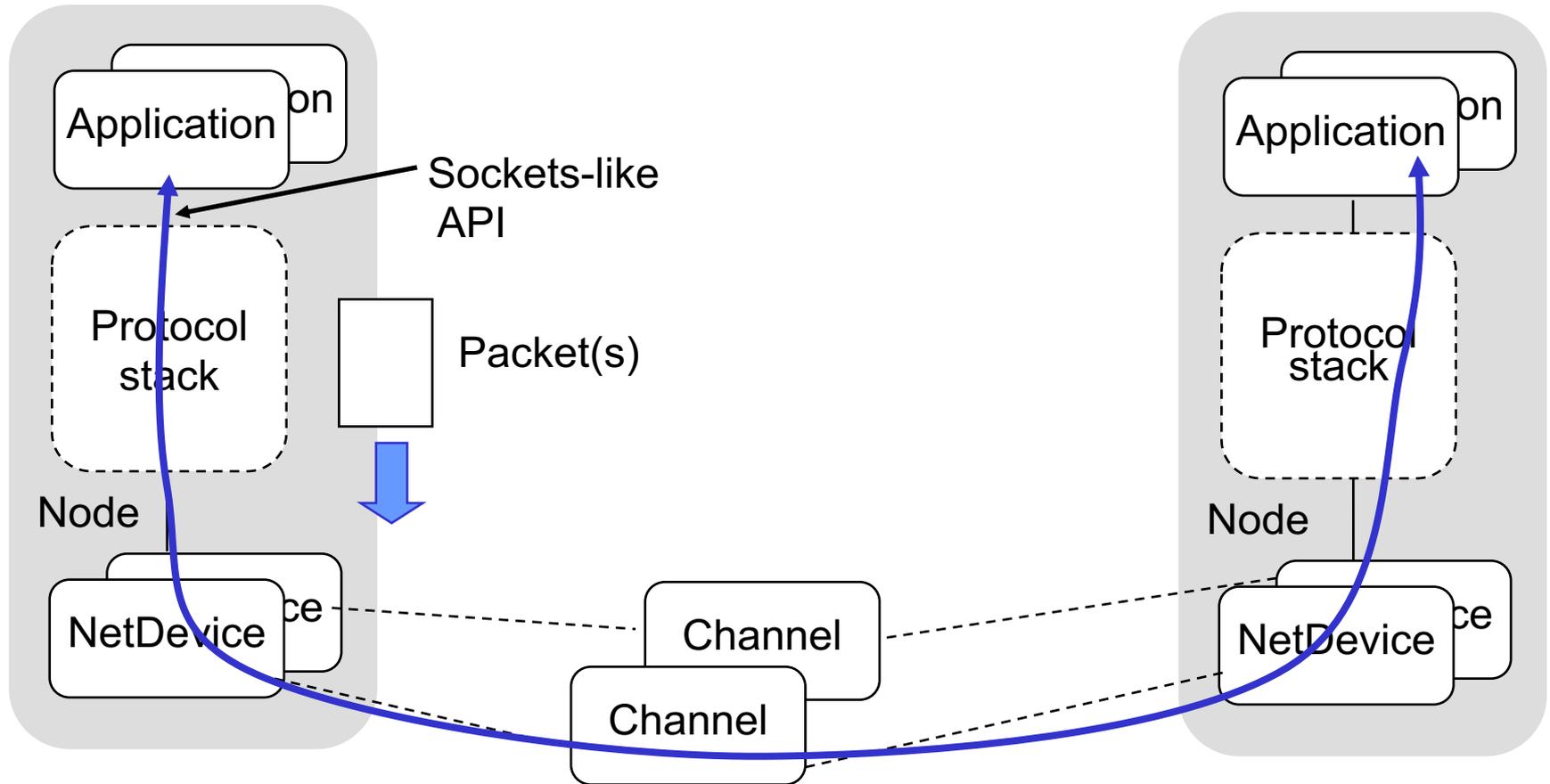
- Main ns-3 website <https://www.nsnam.org>
  - Tutorial, Manual, Model Library
  - Doxygen
  - Wiki
  - Release Notes
  - Training videos from past Workshops on ns-3
- 5G-LENA site: <https://5g-lena.cttc.es/>
  - Getting Started, Manual
  - Doxygen
  - Release Notes

# Placeholder

---

review of udp-client-server.cc example

# Key classes in ns-3



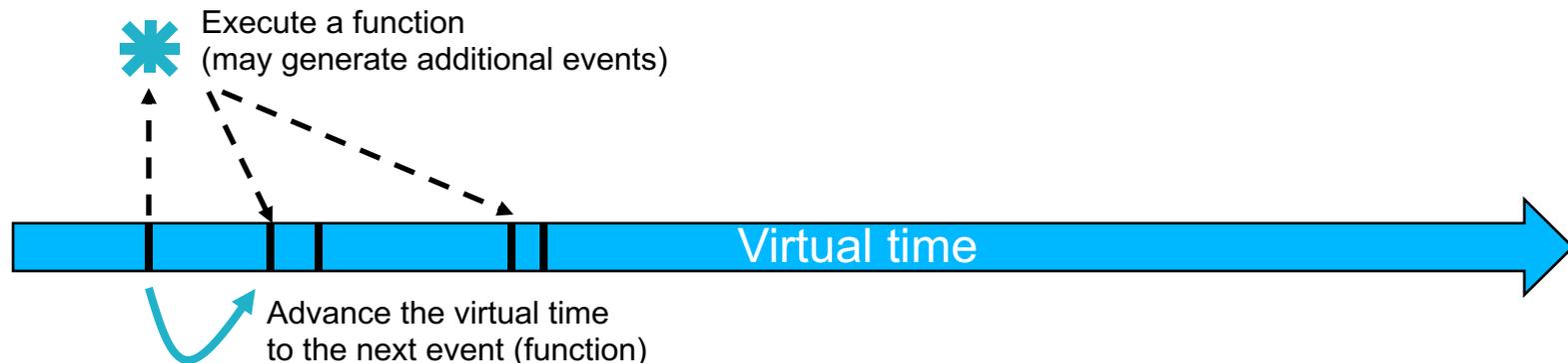
# Key concepts

---

- We are trying to represent the operation of a (wireless) network within a C++ program
- We need a notion of virtual time and of events that occur at specified (virtual) times
- We need a data structure (scheduler) to hold all of these events in temporal order
- We need an object (simulator) to walk the list of events and execute them
- We can choose to ignore things that conceptually occur between our events of interest, focusing only on the (discrete) times with interesting events

# Discrete-event simulation

- Simulation time moves in discrete jumps from event to event
- C++ functions schedule events to occur at specific simulation times
- A simulation scheduler orders the event execution
- Simulation::Run() executes a single-threaded event list
- Simulation stops at specific time or when events end



# ns-3 modules

