Comparison of Polar Decoders with Existing LDPC and Turbo Decoders

Alexios Balatsoukas-Stimming, **Pascal Giard**, and Andreas Burg

Telecommunications Circuits Laboratory
EPFL

IEEE WCNC 2017

Polar codes made it into the next 3GPP wireless communication standard!

Polar codes made it into the next 3GPP wireless communication standard!

- Their theoritical properties render them interesting
  - Explicit construction
  - Provably achieve capacity of some channels
  - Potentially low-complexity encoding and decoding

Polar codes made it into the next 3GPP wireless communication standard!

- Their theoritical properties render them interesting
  - Explicit construction
  - Provably achieve capacity of some channels
  - Potentially low-complexity encoding and decoding

- Their practicality compared to other types of codes has not been thoroughly studied
  - How do they compare against LDPC and turbo codes?
    - In terms of error-correction performance?
    - In terms of hardware efficiency?
    - In terms of decoding latency, energy efficiency, ...?

Polar codes made it into the next 3GPP wireless communication standard!

- Their theoritical properties render them interesting
  - Explicit construction
  - Provably achieve capacity of some channels
  - Potentially low-complexity encoding and decoding

- Their practicality compared to other types of codes has not been thoroughly studied
  - How do they compare against LDPC and turbo codes?
    - In terms of error-correction performance?
    - In terms of hardware efficiency?
    - In terms of decoding latency, energy efficiency, ...?

Polar codes made it into the next 3GPP wireless communication standard!

- Their theoritical properties render them interesting
  - Explicit construction
  - Provably achieve capacity of some channels
  - Potentially low-complexity encoding and decoding

- Their practicality compared to other types of codes has not been thoroughly studied
  - How do they compare against LDPC and turbo codes?
    - In terms of error-correction performance?
    - In terms of hardware efficiency?
    - In terms of decoding latency, energy efficiency, ...?

We humbly attempt to, at least partially, answer these questions.

Comparison Methodology

Quickly, multiple problems arise:

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms
- Many LDPC and turbo codes to choose from!

  $\Rightarrow$ Use codes from well-established communication standards

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms
- Many LDPC and turbo codes to choose from!

  $\Rightarrow$ Use codes from well-established communication standards
- Many important comparison metrics but we have limited time/space!

  $\Rightarrow$ Focus on the two most important, i.e., error-correction performance and hardware efficiency

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms
- Many LDPC and turbo codes to choose from!

  $\Rightarrow$ Use codes from well-established communication standards
- Many important comparison metrics but we have limited time/space!

  $\Rightarrow$ Focus on the two most important, i.e., error-correction performance and hardware efficiency
- How do we make for a fair comparison?! e.g. results are for different error-correction performance, or different CMOS technology.

  $\Rightarrow$ Compare at similar error-correction performance and scale the implementation results using Dennard's laws

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms
- Many LDPC and turbo codes to choose from!

  $\Rightarrow$ Use codes from well-established communication standards
- Many important comparison metrics but we have limited time/space!

  $\Rightarrow$ Focus on the two most important, i.e., error-correction performance and hardware efficiency
- How do we make for a fair comparison?! e.g. results are for different error-correction performance, or different CMOS technology.

  $\Rightarrow$ Compare at similar error-correction performance and scale the implementation results using Dennard's laws

Comparison Methodology

Quickly, multiple problems arise:

- Multiple code constructions and decoding algorithms for polar codes

  $\Rightarrow$ Use the literature where impl. results are presented, i.e., construction with Arıkan's kernel and CRC concat.; SC, BP, and SCL decoding algorithms
- Many LDPC and turbo codes to choose from!

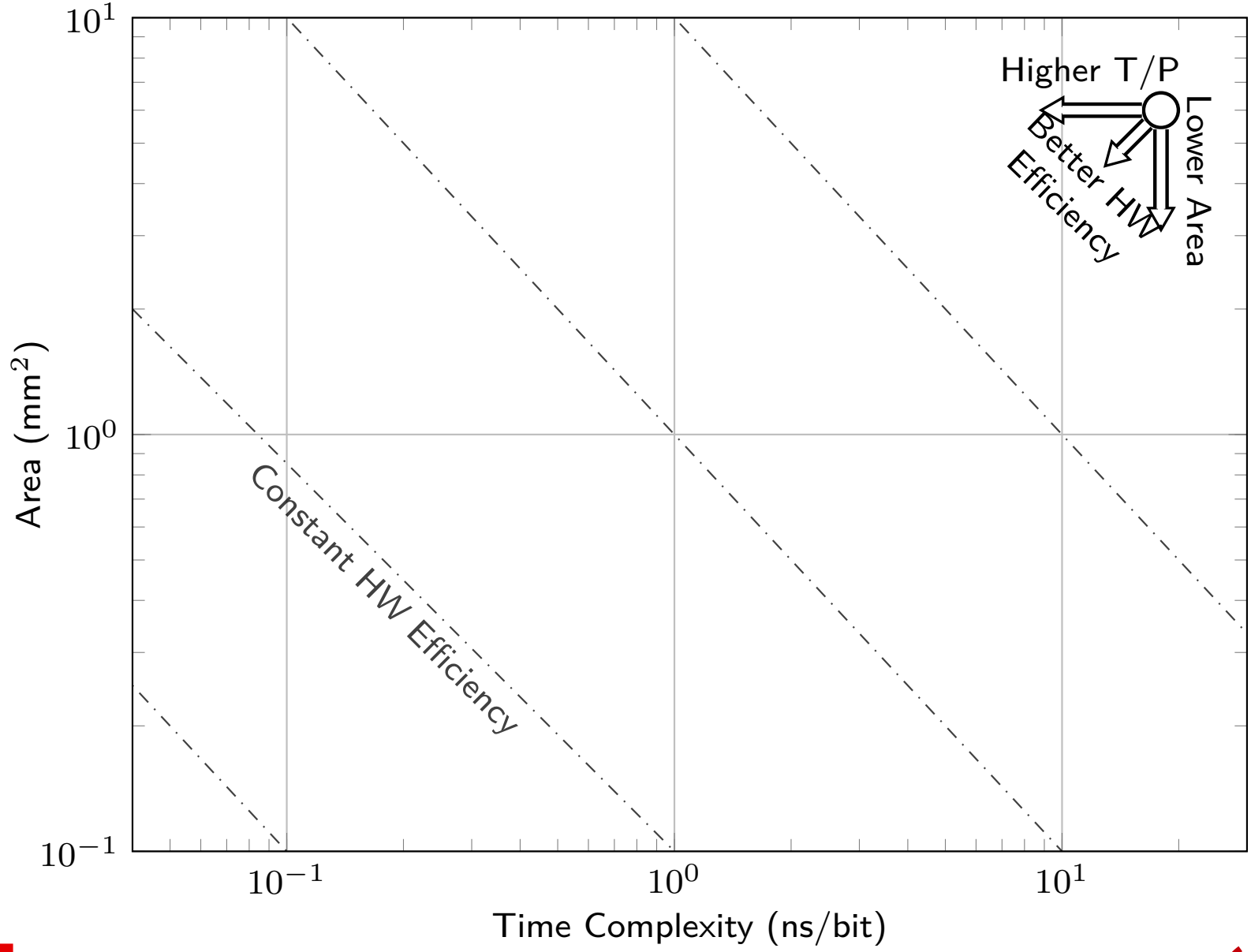
  $\Rightarrow$ Use codes from well-established communication standards
- Many important comparison metrics but we have limited time/space!

  $\Rightarrow$ Focus on the two most important, i.e., error-correction performance and hardware efficiency
- How do we make for a fair comparison?! e.g. results are for different error-correction performance, or different CMOS technology.

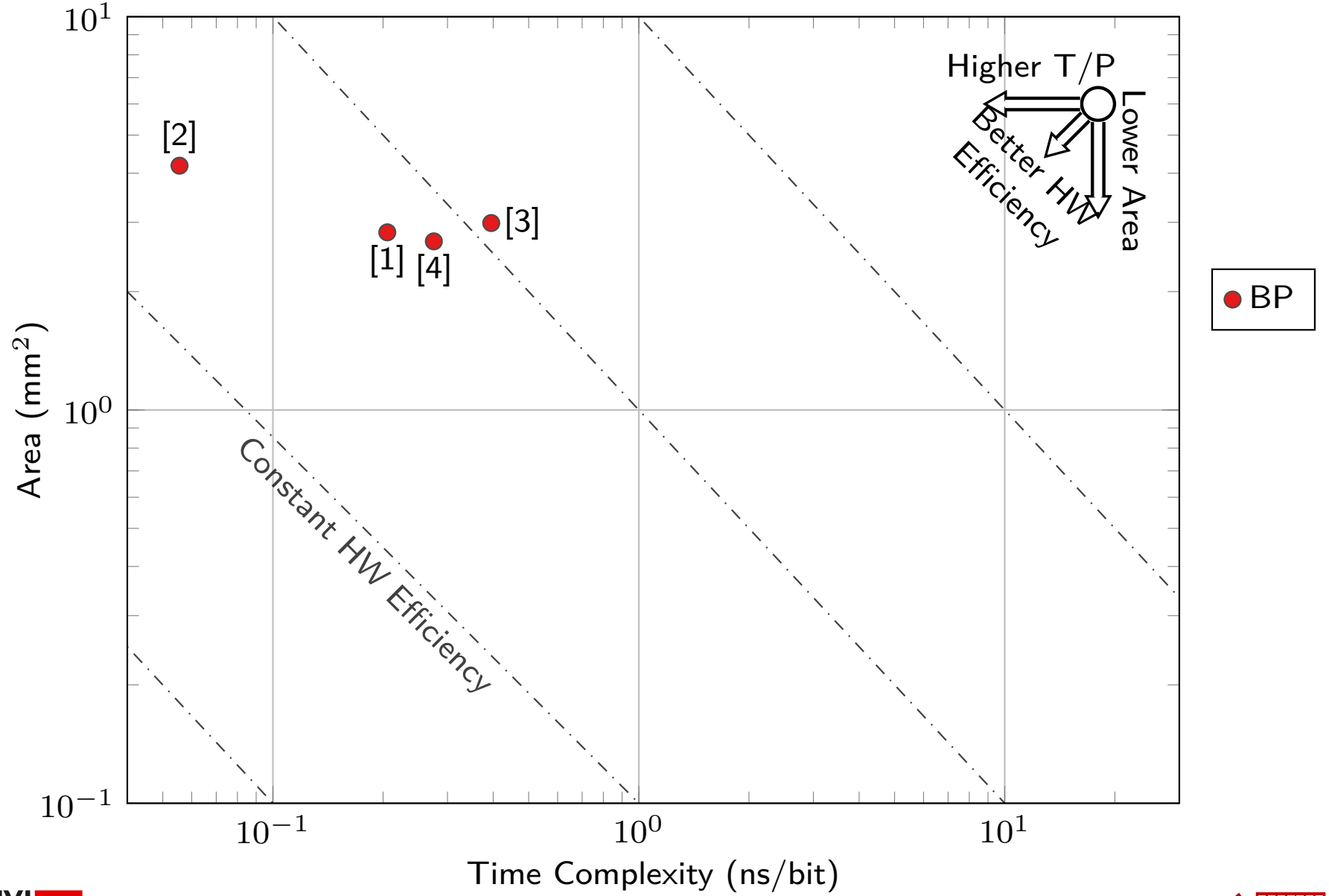
  $\Rightarrow$ Compare at similar error-correction performance and scale the implementation results using Dennard's laws

It remains an "Apples Vs Oranges" comparison but it at least provides a ballpark figure.

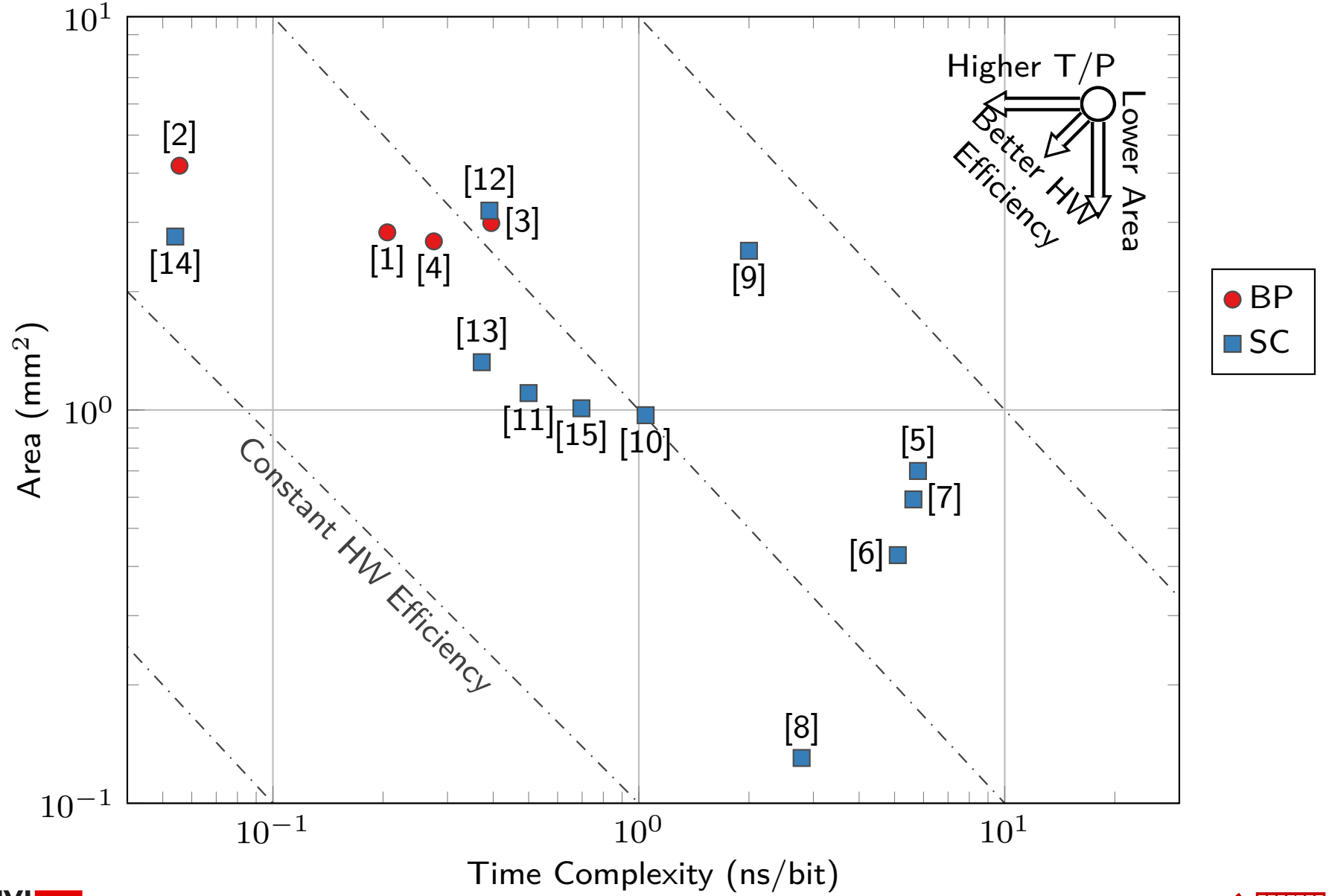
State of Affairs – Polar Decoders from the Lit.



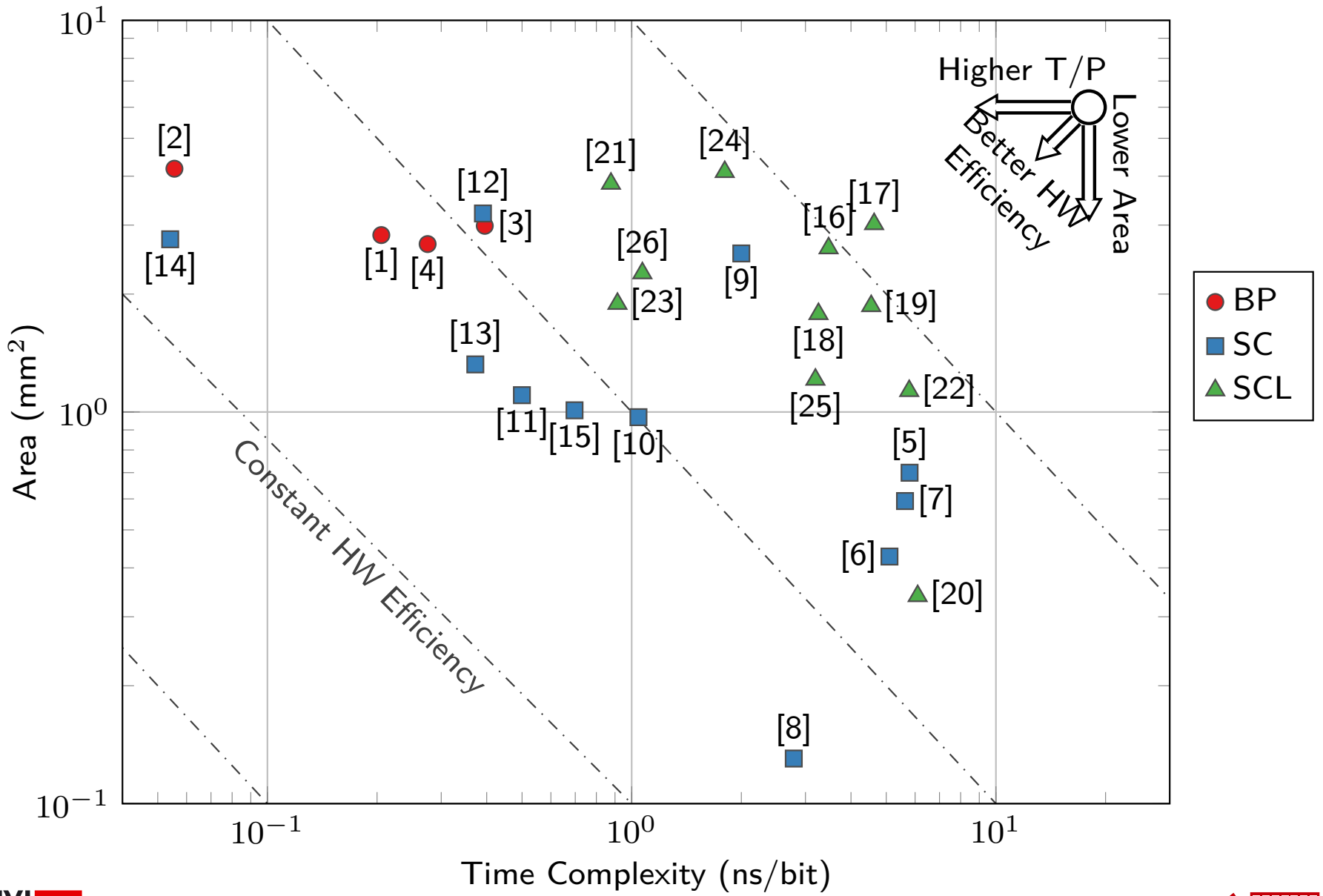
State of Affairs – Polar Decoders from the Lit.



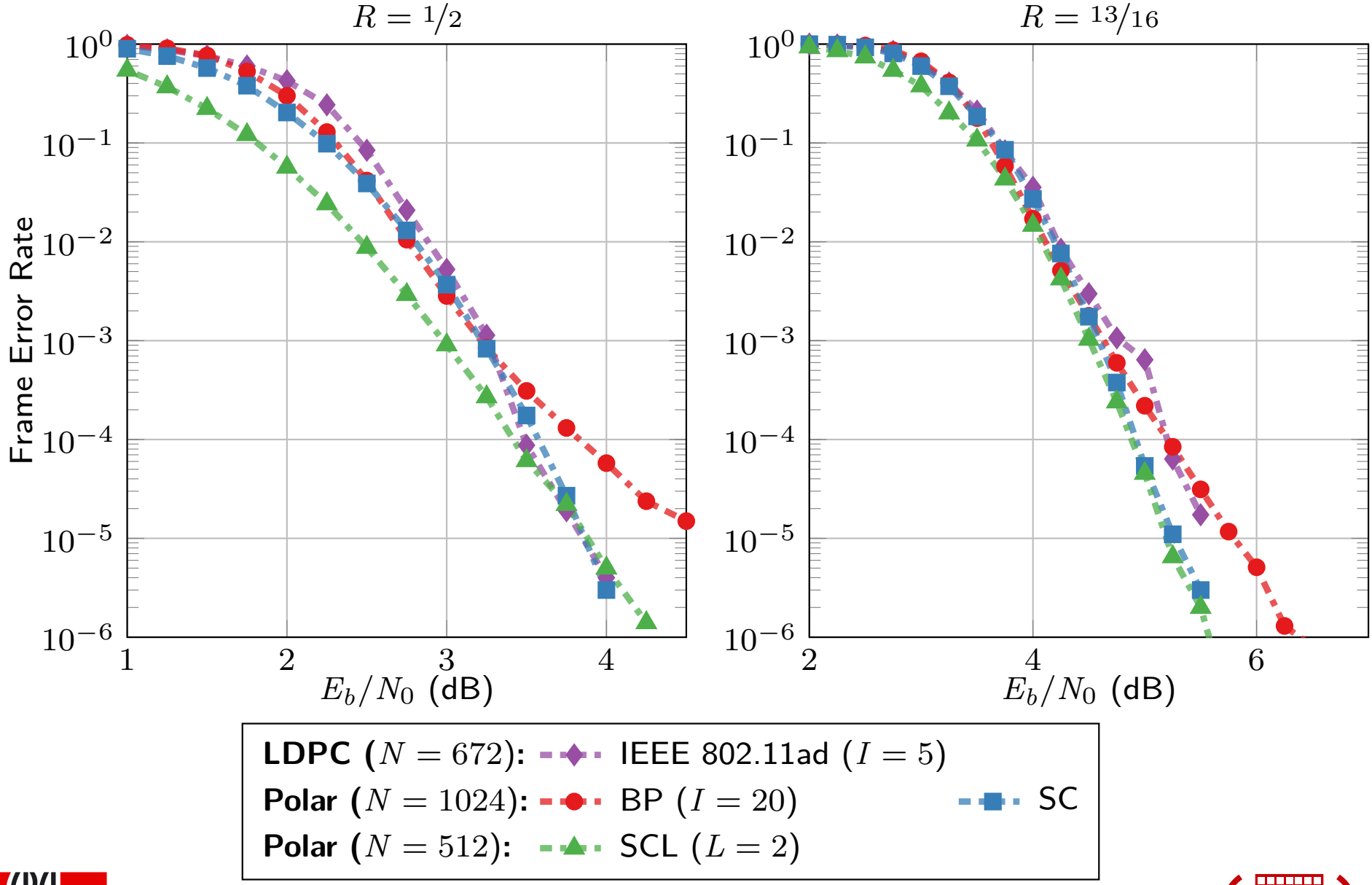
State of Affairs – Polar Decoders from the Lit.



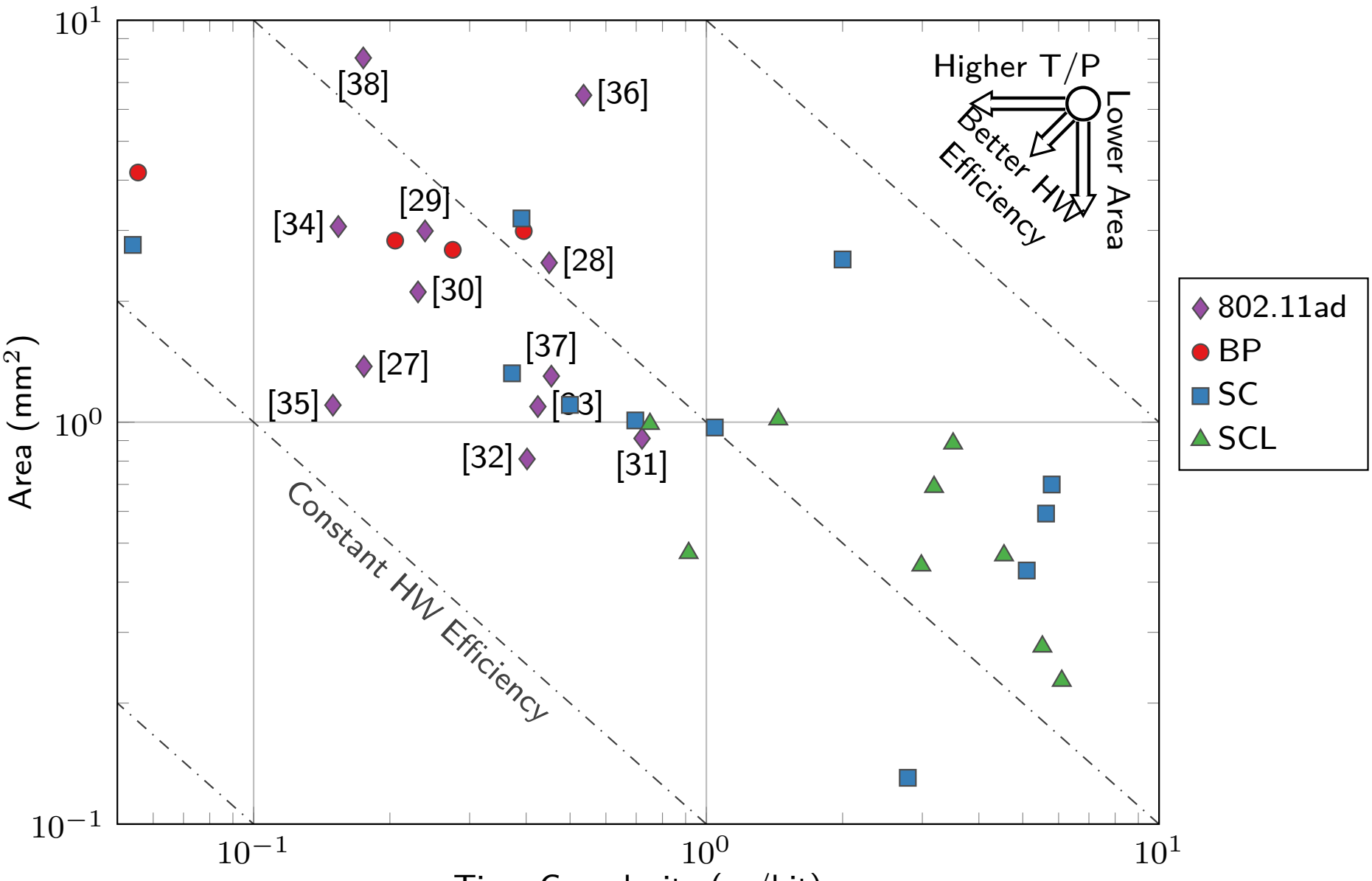
State of Affairs – Polar Decoders from the Lit.



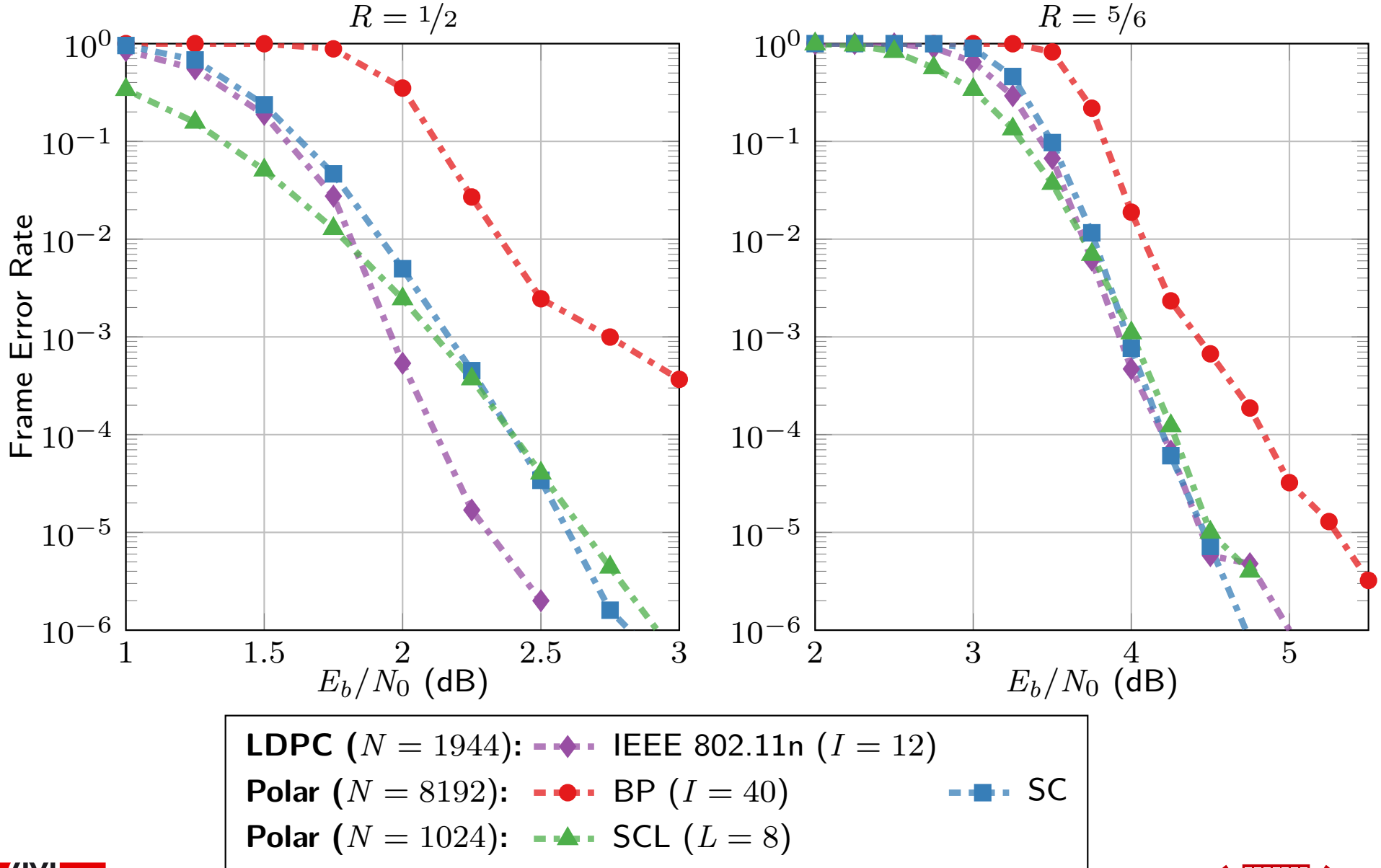
Polar Codes Vs. IEEE 802.11ad LDPC Codes



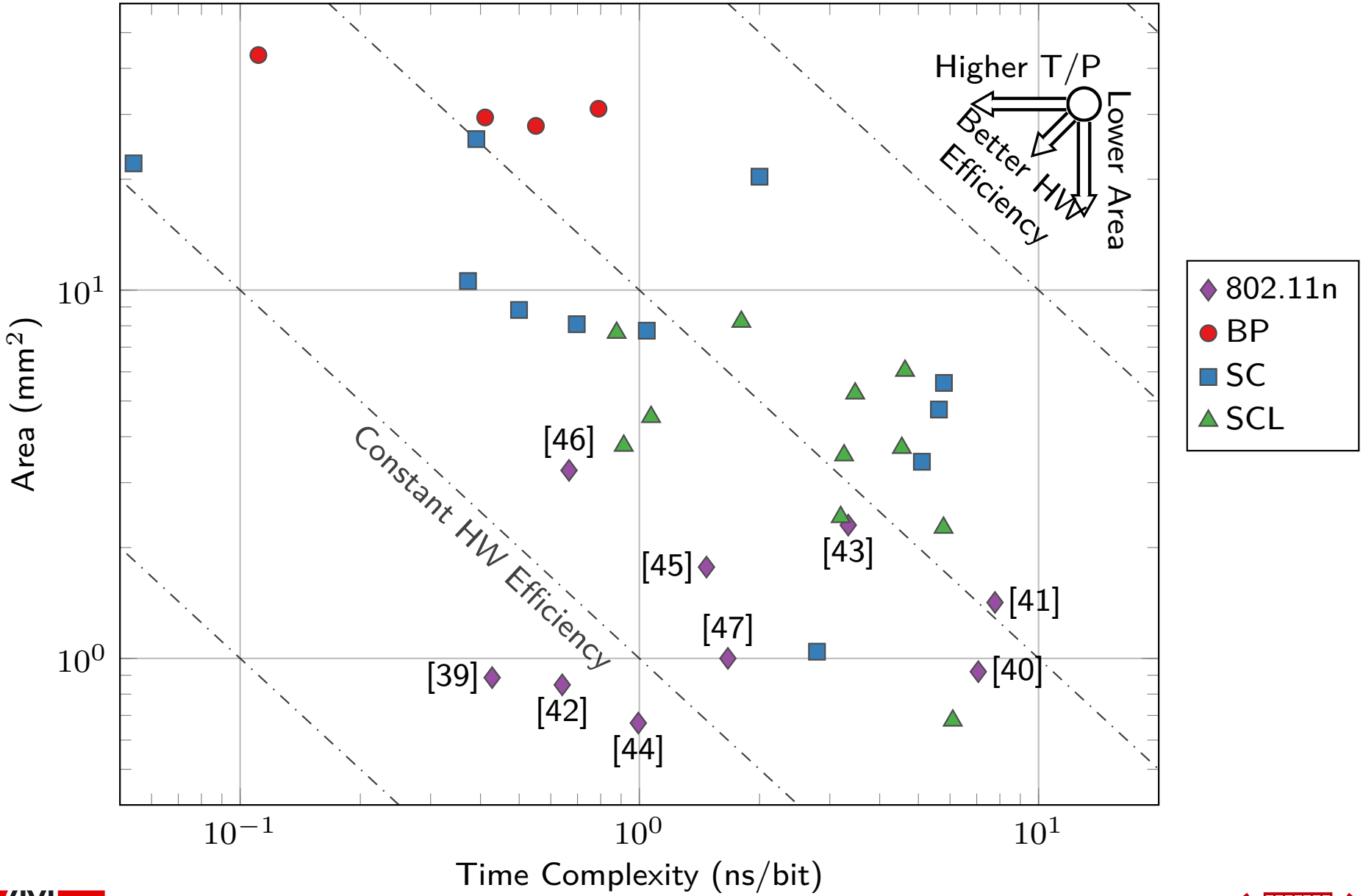
Polar Codes Vs. IEEE 802.11ad LDPC Codes



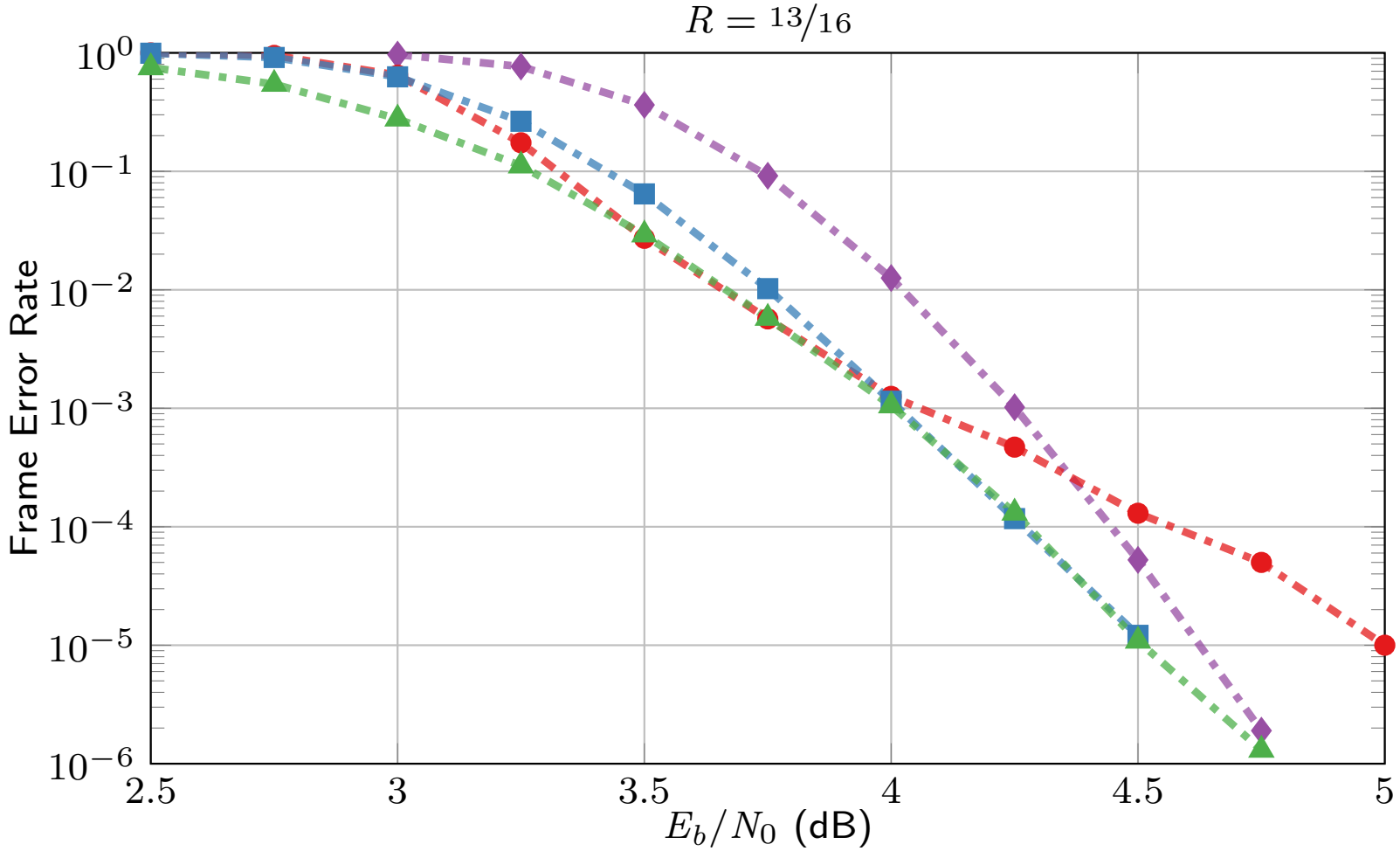
Polar Codes Vs. IEEE 802.11n LDPC Codes



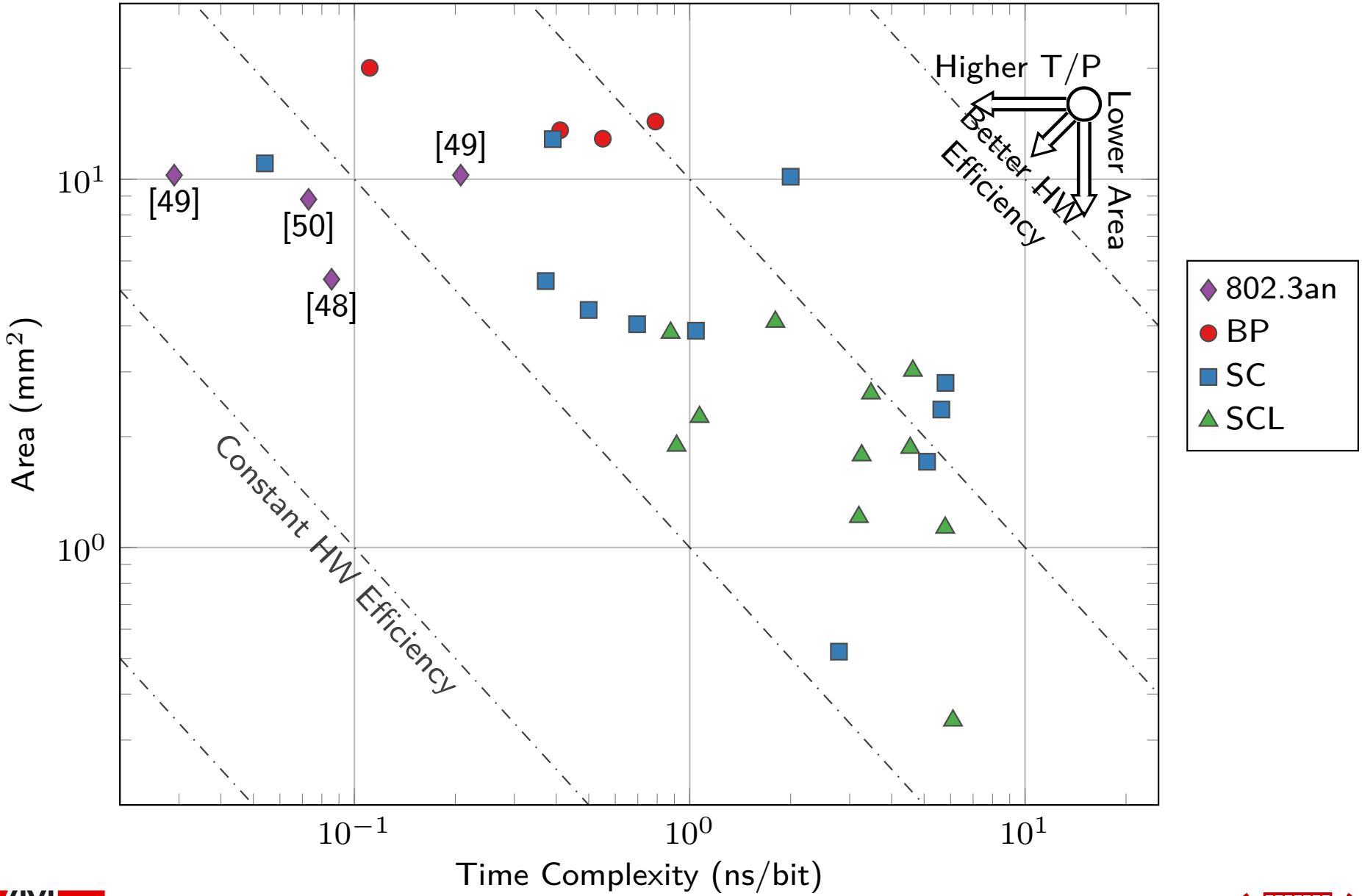
Polar Codes Vs. IEEE 802.11n LDPC Codes



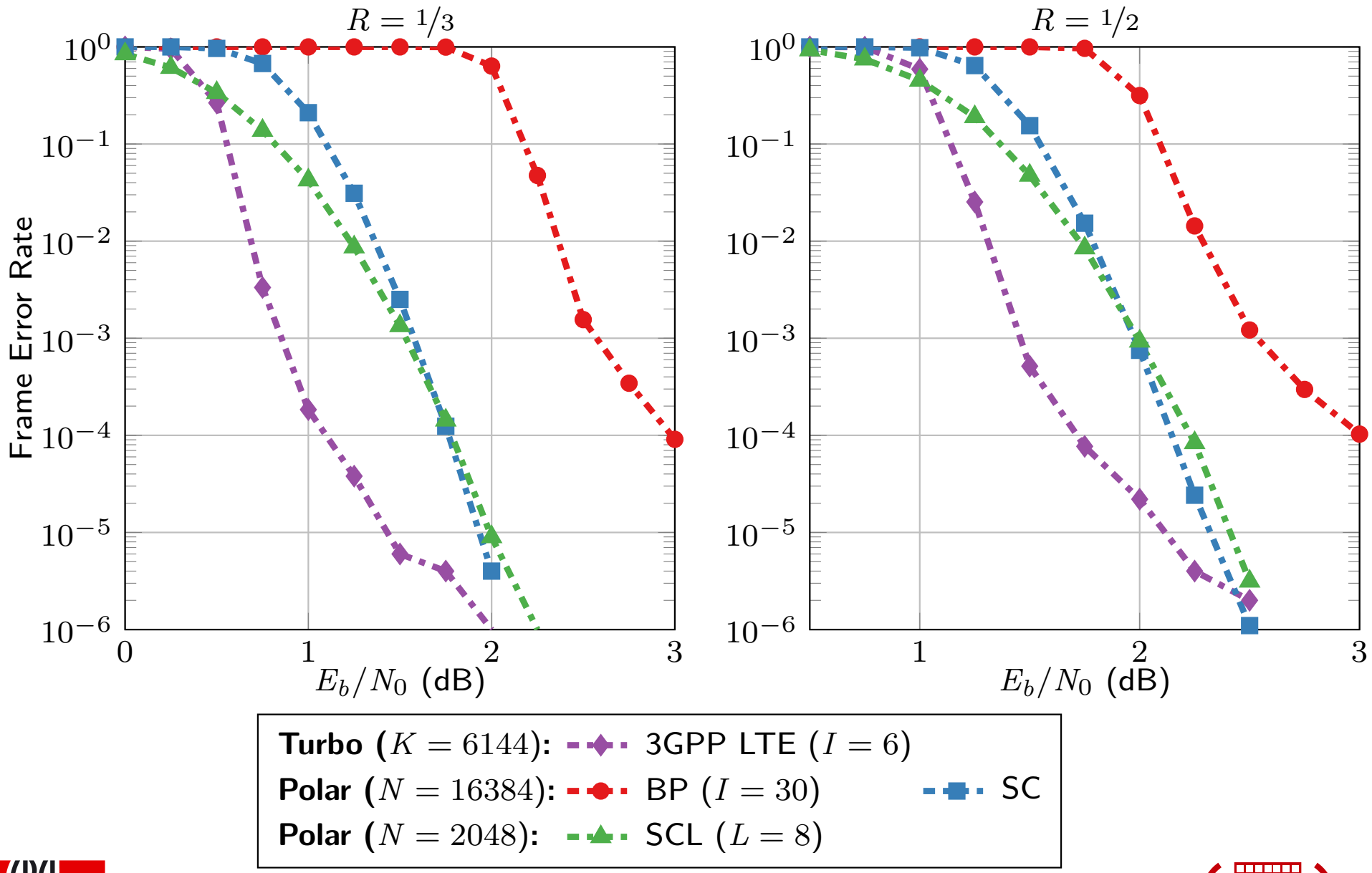
Polar Codes Vs. IEEE 802.3an LDPC Codes



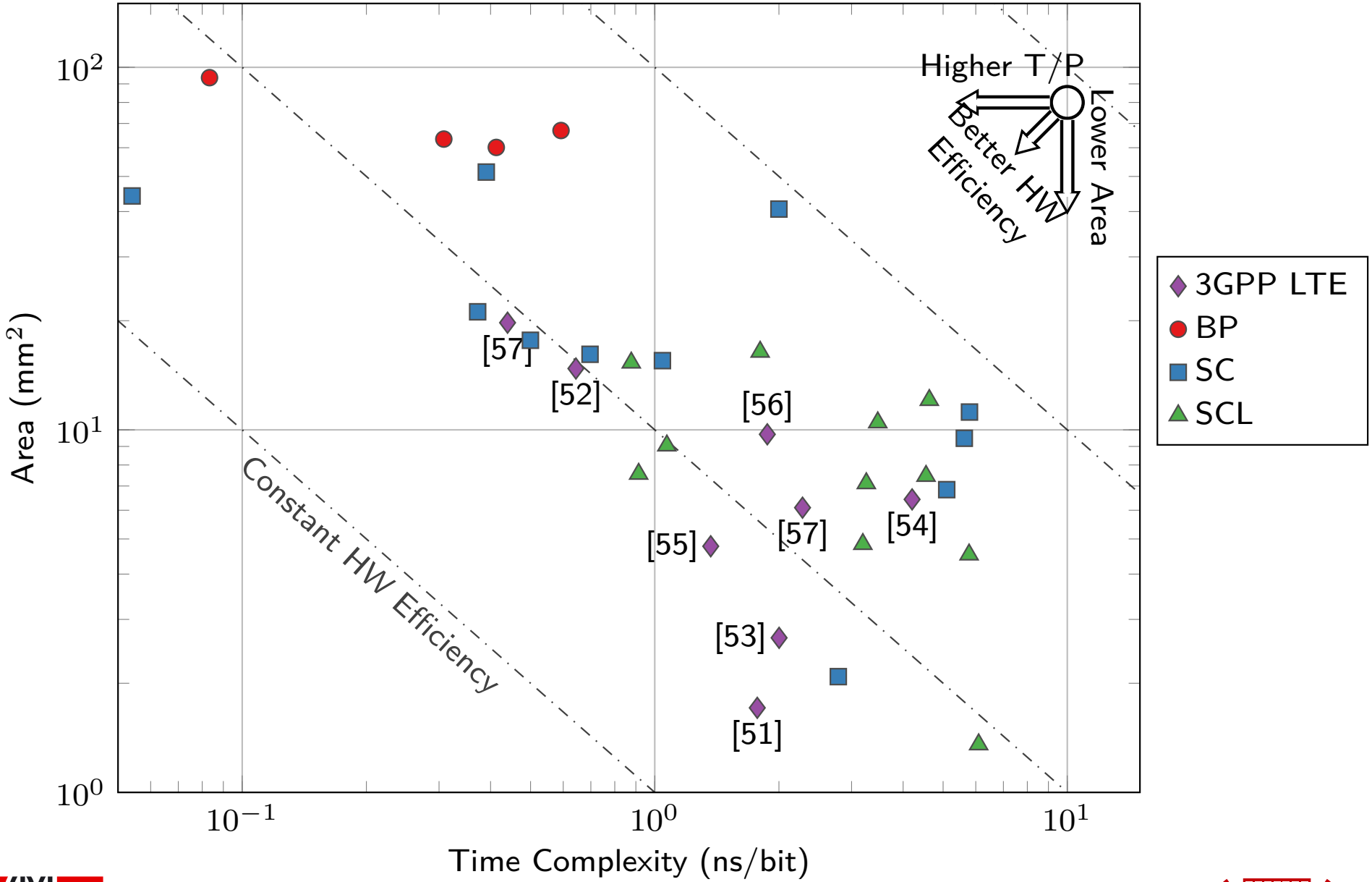
Polar Codes Vs. IEEE 802.3an LDPC Codes



Polar Codes Vs. 3GPP LTE Turbo Codes



Polar Codes Vs. 3GPP LTE Turbo Codes



Summary – Take Away

- Almost impossible to make a fair comparison.

Summary – Take Away

- Almost impossible to make a fair comparison.
- BP and SC decoding often have inferior error-correction performance compared to LDPC/turbo codes.

  $\Rightarrow$ SCL decoding is usually needed to match the performance.

Summary – Take Away

- Almost impossible to make a fair comparison.
- BP and SC decoding often have inferior error-correction performance compared to LDPC/turbo codes.

  $\Rightarrow$ SCL decoding is usually needed to match the performance.
- Hardware efficiency of polar decoders is usually lower than LDPC/turbo counterparts.

  $\Rightarrow$ Significant strides of improvements in the past few years.

Summary – Take Away

- Almost impossible to make a fair comparison.
- BP and SC decoding often have inferior error-correction performance compared to LDPC/turbo codes.

  $\Rightarrow$ SCL decoding is usually needed to match the performance.
- Hardware efficiency of polar decoders is usually lower than LDPC/turbo counterparts.

  $\Rightarrow$ Significant strides of improvements in the past few years.
- Further work required.

  $\Rightarrow$ There's hope as more researchers will turn their attention to polar codes now that they made it into a standard.

Summary – Take Away

- Almost impossible to make a fair comparison.
- BP and SC decoding often have inferior error-correction performance compared to LDPC/turbo codes.

  $\Rightarrow$ SCL decoding is usually needed to match the performance.
- Hardware efficiency of polar decoders is usually lower than LDPC/turbo counterparts.

  $\Rightarrow$ Significant strides of improvements in the past few years.
- Further work required.

  $\Rightarrow$ There's hope as more researchers will turn their attention to polar codes now that they made it into a standard.

Thank you for listening! Questions?

[1] Y. S. Park, Y. Tao, S. Sun, and Z. Zhang, "A 4.68Gb/s belief propagation polar decoder with bit-splitting register file", in *Symp. on VLSI Circ. Dig. of Tech. Papers*, 2014. DOI: 10.1109/VLSIC.2014.6858413.

[2] B. Yuan and K. K. Parhi, "Early stopping criteria for energy-efficient low-latency belief-propagation polar code decoders", *IEEE Trans. Signal Process.*, 62, no., 2014. DOI: 10.1109/TSP.2014.2366712.

[3] S. M. Abbas, Y. Fan, J. Chen, and C.-Y. Tsui, "Low complexity belief propagation polar code decoder", in *IEEE Int. Workshop on Signal Process. Syst. (SiPS)*, 2015. DOI: 10.1109/SiPS.2015.7344986.

[4] S. Sun and Z. Zhang, "Architecture and optimization of high-throughput belief propagation decoding of polar codes", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2016. DOI: 10.1109/ISCAS.2016.7527196.

[5] C. Leroux, A. J. Raymond, G. Sarkis, I. Tal, A. Vardy, and W. J. Gross, "Hardware implementation of successive-cancellation decoders for polar codes", *J. Signal Process. Syst.*, 69, no., 2012. DOI: 10.1007/s11265-012-0685-3.

[6] A. Mishra, A. Raymond, L. Amaru, G. Sarkis, C. Leroux, P. Meinerzhagen, A. Burg, and W. Gross, "A successive cancellation decoder ASIC for a 1024-bit polar code in 180nm CMOS", in *IEEE Asian Solid State Circ. Conf. (A-SSCC)*, 2012. DOI: 10.1109/IPEC.2012.6522661.

[7] C. Leroux, A. Raymond, G. Sarkis, and W. Gross, "A semi-parallel successive-cancellation decoder for polar codes", *IEEE Trans. Signal Process.*, 61, no., 2013, ISSN: 1053-587X. DOI: 10.1109/TSP.2012.2223693.

[8] Y. Fan and C.-Y. Tsui, "An efficient partial-sum network architecture for semi-parallel polar codes decoder implementation", *IEEE Trans. Signal Process.*, 62, no., 2014. DOI: 10.1109/TSP.2014.2319773.

[9] B. Yuan and K. K. Parhi, "Low-latency successive-cancellation polar decoder architectures using 2-bit decoding", *IEEE Trans. Circuits Syst. I*, 61, no., 2014. DOI: 10.1109/TCSI.2013.2283779.

[10] J. Lin, C. Xiong, and Z. Yan, "Reduced complexity belief propagation decoders for polar codes", in *IEEE Int. Workshop on Signal Process. Syst. (SiPS)*, 2015. DOI: 10.1109/SiPS.2015.7344984.

[11] T. Che, J. Xu, and G. Choi, "TC: Throughput centric successive cancellation decoder hardware implementation for polar codes", in *IEEE Int. Conf. on Acoustics, Speech, and Signal Process. (ICASSP)*, 2016. DOI: 10.1109/ICASSP.2016.7471824.

[12] O. Dizdar and E. Arıkan, "A high-throughput energy-efficient implementation of successive cancellation decoder for polar codes using combinational logic", *IEEE Trans. Circuits Syst. I*, 63, no., 2016. DOI: 10.1109/TCSI.2016.2525020.

[13] P. Giard, A. Balatsoukas-Stimming, G. Sarkis, C. Thibeault, and W. J. Gross, "Fast low-complexity decoders for low-rate polar codes", *J. Signal Process. Syst.*, PP, no., 2016. DOI: 10.1007/s11265-016-1173-y.

[14] P. Giard, G. Sarkis, C. Thibeault, and W. J. Gross, "Multi-mode unrolled hardware architectures for polar decoders", *IEEE Trans. Circuits Syst. I*, 63, no., 2016, ISSN: 1549-8328. DOI: 10.1109/TCSI.2016.2586218.

[15] J. Lin, J. Sha, L. Li, C. Xiong, Z. Yan, and Z. Wang, "A high throughput belief propagation decoder architecture for polar codes", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2016. DOI: 10.1109/ISCAS.2016.7527193.

[16] A. Balatsoukas-Stimming, A. J. Raymond, W. J. Gross, and A. Burg, "Hardware architecture for list successive cancellation decoding of polar codes", *IEEE Trans. Circuits Syst. II*, 61, no., 2014. DOI: 10.1109/TCSII.2014.2327336.

[17] J. Lin and Z. Yan, "Efficient list decoder architecture for polar codes", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2014. DOI: 10.1109/ISCAS.2014.6865312.

[18] A. Balatsoukas-Stimming, M. Bastani Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes", *IEEE Trans. Signal Process.*, 63, no., 2015, ISSN: 1053-587X. DOI: 10.1109/TSP.2015.2439211.

[19] Y. Fan, J. Chen, C. Xia, C.-Y. Tsui, J. Jin, H. Shen, and B. Li, "Low-latency list decoding of polar codes with double thresholding", in *IEEE Int. Conf. on Acoustics, Speech, and Signal Process. (ICASSP)*, 2015. DOI: 10.1109/ICASSP.2015.7178128.

[20] S. A. Hashemi, A. Balatsoukas-Stimming, P. Giard, C. Thibeault, and W. J. Gross, "Partitioned successive-cancellation list decoding of polar codes", in *IEEE Int. Conf. on Acoustics, Speech, and Signal Process. (ICASSP)*, 2016. DOI: 10.1109/ICASSP.2016.7471817.

[21] J. Lin, C. Xiong, and Z. Yan, "A high throughput list decoder architecture for polar codes", *IEEE Trans. VLSI Syst.*, 24, no., 2016. DOI: 10.1109/TVLSI.2015.2499777.

[22] J. Lin and Z. Yan, "An efficient list decoder architecture for polar codes", *IEEE Trans. VLSI Syst.*, 23, no., 2015. DOI: 10.1109/TVLSI.2014.2378992.

[23] C. Xiong, J. Lin, and Z. Yan, "A multimode area-efficient SCL polar decoder", *IEEE Trans. VLSI Syst.*, PP, no., 2016. DOI: 10.1109/TVLSI.2016.2557806.

[24] B. Yuan and K. K. Parhi, "Low-latency successive-cancellation list decoders for polar codes with multibit decision", *IEEE Trans. VLSI Syst.*, 23, no., 2015, ISSN: 1063-8210. DOI: 10.1109/TVLSI.2014.2359793.

[25] C. Xiong, J. Lin, and Z. Yan, "Symbol-decision successive cancellation list decoder for polar codes", *IEEE Trans. Signal Process.*, 64, no., 2016. DOI: 10.1109/TSP.2015.2486750.

[26] B. Yuan and K. K. Parhi, "LLR-based successive-cancellation list decoder for polar codes with multi-bit decision", *IEEE Trans. Circuits Syst. II*, PP, no., 2016. DOI: 10.1109/TCSII.2016.2546904.

[27] H. Shirani-Mehr, T. Mohsenin, and B. Baas, "A reduced routing network architecture for partial parallel LDPC decoders", in *Asilomar Conf. on Signals, Syst. and Computers*, 2011. DOI: 10.1109/ACSSC.2011.6190420.

[28] M. Weiner, B. Nikolic, and Z. Zhang, "LDPC decoder architecture for high-data rate personal-area networks", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2011. DOI: 10.1109/ISCAS.2011.5937930.

[29] S.-W. Yen, S.-Y. Hung, C.-L. Chen, H.-C. Chang, S.-J. Jou, and C.-Y. Lee, "A 5.79-Gb/s energy-efficient multirate LDPC codec chip for IEEE 802.15.3c applications", *IEEE J. Solid-State Circuits*, 47, no., 2012. DOI: 10.1109/JSSC.2012.2194176.

[30] S. Ajaz and H. Lee, "Reduced-complexity local switch based multi-mode QC-LDPC decoder architecture for Gbit wireless communication", *Electron. Lett.*, 49, no., 2013. DOI: 10.1049/el.2013.1673.

[31] A. Balatsoukas-Stimming, N. Preyss, A. Cevrero, A. Burg, and C. Roth, "A parallelized layered QC-LDPC decoder for IEEE 802.11ad", in *IEEE Int. New Circ. and Syst. Conf. (NEWCAS)*, 2013. DOI: 10.1109/NEWCAS.2013.6573590.

[32] M. Li, F. Naessens, P. Debacker, P. Raghavan, C. Desset, M. Li, A. Dejonghe, and L. Van der Perre, "An area and energy efficient half-row-paralleled layer LDPC decoder for the 802.11ad standard", in *IEEE Int. Workshop on Signal Process. Syst. (SiPS)*, 2013. DOI: 10.1109/SiPS.2013.6674490.

[33] M. Li, F. Naessens, M. Li, P. Debacker, C. Desset, P. Raghavan, A. Dejonghe, and L. Van der Perre, "A processor based multi-standard low-power LDPC engine for multi-Gbps wireless communication", in *IEEE Glob. Conf. on Signal and Inf. Process. (GlobalSIP)*, 2013. DOI: 10.1109/GlobalSIP.2013.6737136.

[34] Y. S. Park, D. Blaauw, D. Sylvester, and Z. Zhang, "Low-power high-throughput LDPC decoder using non-refresh embedded DRAM", *IEEE J. Solid-State Circuits*, 49, no., 2014. DOI: 10.1109/JSSC.2014.2300417.

[35] S. Ajaz and H. Lee, "Multi-Gb/s multi-mode LDPC decoder architecture for IEEE 802.11ad standard", in *IEEE Asia Pacific Conf. on Circ. and Syst. (APCCAS)*, 2014. DOI: 10.1109/APCCAS.2014.7032742.

[36] M. Weiner, M. Blagojevic, S. Skotnikov, A. Burg, P. Flatresse, and B. Nikolic, "A scalable 1.5-to-6Gb/s 6.2-to-38.1mW LDPC decoder for 60GHz wireless networks in 28nm UTBB FDSOI", in *IEEE Int. Solid-State Circ. Conf. (ISSCC)*, 2014. DOI: 10.1109/ISSCC.2014.6757515.

[37] M. Li, Y. Lee, Y. Huang, and L. Van der Perre, "Area and energy efficient 802.11ad LDPC decoding processor", *Electron. Lett.*, 51, no., 2015. DOI: 10.1049/el.2014.4263.

[38] M. Li, J. W. Weijers, V. Derudder, I. Vos, M. Rykunov, S. Dupont, P. Debacker, A. Dewilde, Y. Huang, L. V. d. Perre, and W. V. Thillo, "An energy efficient 18Gbps LDPC decoding processor for 802.11ad in 28nm CMOS", in *IEEE Asian Solid-State Circ. Conf. (A-SSCC)*, 2015. DOI: 10.1109/ASSCC.2015.7387473.

[39] K. Gunnam, G. Choi, W. Wang, and M. Yeary, "Multi-rate layered decoder architecture for block LDPC codes of the IEEE 802.11n wireless standard", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2007. DOI: 10.1109/ISCAS.2007.378835.

[40] M. Rovini, G. Gentile, F. Rossi, and L. Fanucci, "A minimum-latency block-serial architecture of a decoder for IEEE 802.11n LDPC codes", in *IFIP Int. Conf. on Very Large Scale Integration*, 2007. DOI: 10.1109/VLSISOC.2007.4402504.

[41] M. Rovini, G. Gentile, F. Rossi, and L. Fanucci, "A scalable decoder architecture for IEEE 802.11n LDPC codes", in *IEEE Glob. Telecommun. Conf. (GLOBECOM)*, 2007. DOI: 10.1109/GLOCOM.2007.620.

[42] C. Studer, N. Preyss, C. Roth, and A. Burg, "Configurable high-throughput decoder architecture for quasi-cyclic LDPC codes", in *Asilomar Conf. on Signals, Syst. and Computers*, 2008. DOI: 10.1109/ACSSC.2008.5074592.

[43] Y. Sun, J. R. Cavallaro, and T. Ly, "Scalable and low power LDPC decoder design using high level algorithmic synthesis", in *IEEE Int. SOC Conf. (SOCC)*, 2009. DOI: 10.1109/SOCCON.2009.5398044.

[44] J. Jin and C. y. Tsui, "An energy efficient layered decoding architecture for LDPC decoder", *IEEE Trans. VLSI Syst.*, 18, no., 2010. DOI: 10.1109/TVLSI.2009.2021479.

[45] C. Roth, P. Meinerzhagen, C. Studer, and A. Burg, "A 15.8 pJ/bit/iter quasi-cyclic LDPC decoder for IEEE 802.11n in 90 nm CMOS", in *IEEE Asian Solid State Circ. Conf. (A-SSCC)*, 2010. DOI: 10.1109/ASSCC.2010.5716618.

[46] Y. Sun, G. Wang, and J. R. Cavallaro, "Multi-layer parallel decoding algorithm and VLSI architecture for quasi-cyclic LDPC codes", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2011. DOI: 10.1109/ISCAS.2011.5937928.

[47] P. Meinerzhagen, A. Bonetti, G. Karakonstantis, C. Roth, F. Gürkaynak, and A. Burg, "Refresh-free dynamic standard-cell based memories: Application to a QC-LDPC decoder", in *IEEE Int. Symp. on Circ. and Syst. (ISCAS)*, 2015. DOI: 10.1109/ISCAS.2015.7168911.

[48] A. Cevrero, Y. Leblebici, P. Ienne, and A. Burg, "A 5.35 mm^2 10GBASE-T Ethernet LDPC decoder chip in 90 nm CMOS", in *IEEE Asian Solid State Circ. Conf. (A-SSCC)*, 2010. DOI: 10.1109/ASSCC.2010.5716619.

[49] Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, "An efficient 10GBASE-T Ethernet LDPC decoder design with low error floors", *IEEE J. Solid-State Circuits*, 45, no., 2010. DOI: 10.1109/JSSC.2010.2042255.

[50] D. Bao, X. Chen, Y. Huang, C. Wu, Y. Chen, and X. Y. Zeng, "A single-routing layered LDPC decoder for 10GBASE-T Ethernet in 130nm CMOS", in *Asia and South Pacific Design Autom. Conf. (ASP-DAC)*, 2012. DOI: 10.1109/ASPDAC.2012.6165020.

[51] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and implementation of a parallel turbo-decoder ASIC for 3GPP-LTE", *IEEE J. Solid-State Circuits*, 46, no., 2011. DOI: 10.1109/JSSC.2010.2075390.

[52] T. IInseher, F. Kienle, C. Weis, and N. Wehn, "A 2.15GBit/s turbo code decoder for LTE advanced base station applications", in *Int. Symp. on Turbo Codes and Iterative Inf. Process. (ISTC)*, 2012. DOI: 10.1109/ISTC.2012.6325191.

[53] X. Chen, Y. Chen, Y. Li, Y. Huang, and X. Zeng, "A 691 Mbps 1.392mm^2 configurable radix-16 turbo decoder ASIC for 3GPP-LTE and WiMAX systems in 65nm CMOS", in *IEEE Asian Solid-State Circ. Conf. (A-SSCC)*, 2013. DOI: 10.1109/ASSCC.2013.6691006.

[54] C.-Y. Lin, C.-C. Wong, and H.-C. Chang, "A 40 nm 535 Mbps multiple code-rate turbo decoder chip using reciprocal dual trellis", *IEEE J. Solid-State Circuits*, 48, no., 2013. DOI: 10.1109/JSSC.2013.2274883.

[55] S. Belfanti, C. Roth, M. Gautschi, C. Benkeser, and Q. Huang, "A 1Gbps LTE-advanced turbo-decoder ASIC in 65nm CMOS", in *Symp. on VLSI Circ. (VLSIC)*, 2013.

[56] R. Shrestha and R. Paily, "High-throughput turbo decoder with parallel architecture for LTE wireless communication standards", *IEEE Trans. Circuits Syst. I*, 61, no., 2014. DOI: 10.1109/TCSI.2014.2332266.

[57] G. Wang, H. Shen, Y. Sun, J. Cavallaro, A. Vosoughi, and Y. Guo, "Parallel interleaver design for a high throughput HSPA+/LTE multi-standard turbo decoder", *IEEE Trans. Circuits Syst. I*, 61, no., 2014. DOI: 10.1109/TCSI.2014.2309810.

Bonus Slides

SC Decoder Hardware Implementations

Work	Main Contribution	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	T/P (Mb/s)	Power (mW)
Leroux *et al.* [5]	Linear memory requirement.	1024	65	0.36	500	239	n/a
Mishra *et al.* [6]	First ASIC, two-bit decoding.	1024	180	1.71	150	98	67
Leroux *et al.* [7]	Semi-parallel architecture.	1024	65	0.31	500	246	n/a
Fan *et al.* [8]	Efficient partial sum computation.	1024	65	0.07	1010	497	n/a
Yuan *et al.* [9]	Two-bit decoding with precomputation.	1024	45	0.64	750	1000	n/a
Lin *et al.* [10]	SCAN decoder.	1024	90	0.97	571	958	n/a
Che *et al.* [11]	Implementation of fast-SSC decoding.	1024	45	0.28	1040	4004	n/a
Dizdar *et al.* [12]	Single-cycle implementation.	1024	90	3.21	2	2560	191
Giard *et al.* [13]	Fast-SSC decoding for low-rate codes.	1024	65	0.69	600	3720	215
Giard *et al.* [14]	Unrolled decoder implementation.	1024	65	1.44	650	25600	546
Lin *et al.* [15]	Fast-SSC SCAN decoder.	1024	90	1.01	471	1435	n/a

BP Decoder Hardware Implementations

Work	Main Contribution	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	Max. Iter.	Max. T/P (Mb/s)	Sust. T/P (Mb/s)	Power (mW)
Park *et al.* [1]	Unidirectional schedule.	1024	65	1.48	300	15	4676	2048	478
Yuan *et al.* [2]	Early termination.	1024	45	1.04	500	40	4500	2588	990
Abbas *et al.* [3]	Sub-graph freezing.	1024	45	0.75	197	15	1683	1683	n/a
Lin *et al.* [4]	Adaptive quantization.	1024	65	1.40	769	5	7870	7870	442

SCL Decoder Hardware Implementations ($L = 4$)

Work	Main Contribution	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	T/P (Mb/s)
Balatsoukas-Stimming *et al.* [16]	First architecture, efficient path copying.	1024	90	2.62	730	288
Lin *et al.* [17]	First CRC-aided decoder.	1024	90	3.02	657	216
Balatsoukas-Stimming *et al.* [18]	LLR-based implementation.	1024	90	1.78	794	307
Fan *et al.* [19]	Low-complexity path metric sorting.	1024	90	n/a	n/a	n/a
Hashemi *et al.* [20]	Reduced memory requirements.	2048	90	1.36	500	164
Lin *et al.* [22]	Efficient memory and metric sorting.	1024	90	1.13	476	173
Lin *et al.* [21]	Fast-SSC decoding with approximations.	1024	90	3.83	403	1140
Xiong *et al.* [23]	Partial ML decoding, flexibility.	1024	90	1.89	409	1094
Yuan *et al.* [24]	Multi-bit decision LL-based decoding.	1024	65	2.14	400	401
Xiong *et al.* [25]	Symbol-based SCL decoding.	1024	90	1.21	500	313
Yuan *et al.* [26]	Multi-bit decision LLR-based decoding.	1024	65	1.18	360	675

IEEE 802.11ad LDPC Decoder Implementations

Work	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	Schedule	Iter.	T/P (Mb/s)	Voltage (V)	Power (mw)
Shrirani-Mehr *et al.* [27]	672	65	0.72	235	Layered	5	7900	n/a	n/a
Weiner *et al.* [28]	672	65	1.30	150	Flooding	15	3080	0.8	84
Yen *et al.* [29]	672	65	1.56	197	Layered	5	5790	1.0	361
Ajaz *et al.* [30]	672	65	1.10	215	Layered	6	6000	1.1	210
Balatsoukas-Stimming *et al.* [31]	672	40	0.18	850	Layered	5	3120	n/a	n/a
Li *et al.* [32]	672	40	0.16	500	Layered	5	5600	0.9	99
Li *et al.* [33]	672	40	0.22	500	Layered	5	5300	0.9	136
Park *et al.* [34]	672	65	1.60	540	Flooding	10	9000	1.1	783
Ajaz *et al.* [35]	672	65	0.57	400	Layered	7	9250	1.1	273
Weiner *et al.* [36]	672	28	0.63	65	Flooding	15	6000	0.7	38
Li *et al.* [37]	672	28	0.13	400	Layered	3	7070	0.8	54
Li *et al.* [38]	672	28	0.78	470	Layered	5	18400	0.9	166

IEEE 802.11n LDPC Decoder Implementations

Work	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	Schedule	Iter.	T/P (Mb/s)	Voltage (V)	Power (mW)
Gunnam *et al.* [39]	1944	130	1.85	500	Layered	15	1618	n/a	238
Rovini *et al.* [40]	1944	65	0.48	240	Layered	12	196	1.2	168
Rovini *et al.* [41]	1944	65	0.74	240	Layered	12	178	1.2	234
Studer *et al.* [42]	1944	180	3.39	208	Layered	5	780	n/a	2886
Sun *et al.* [43]	2304	65	1.20	400	Layered	10	415	0.9	180
Jin *et al.* [44]	1944	180	2.67	250	Layered	10	503	1.8	463
Roth *et al.* [45]	1944	90	1.77	346	Layered	10	679	1.0	107
Sun *et al.* [46]	1944	45	0.81	815	Layered	15	3000	n/a	n/a
Meinerzhagen *et al.* [47]	1944	90	1.00	307	Layered	10	600	1.0	88

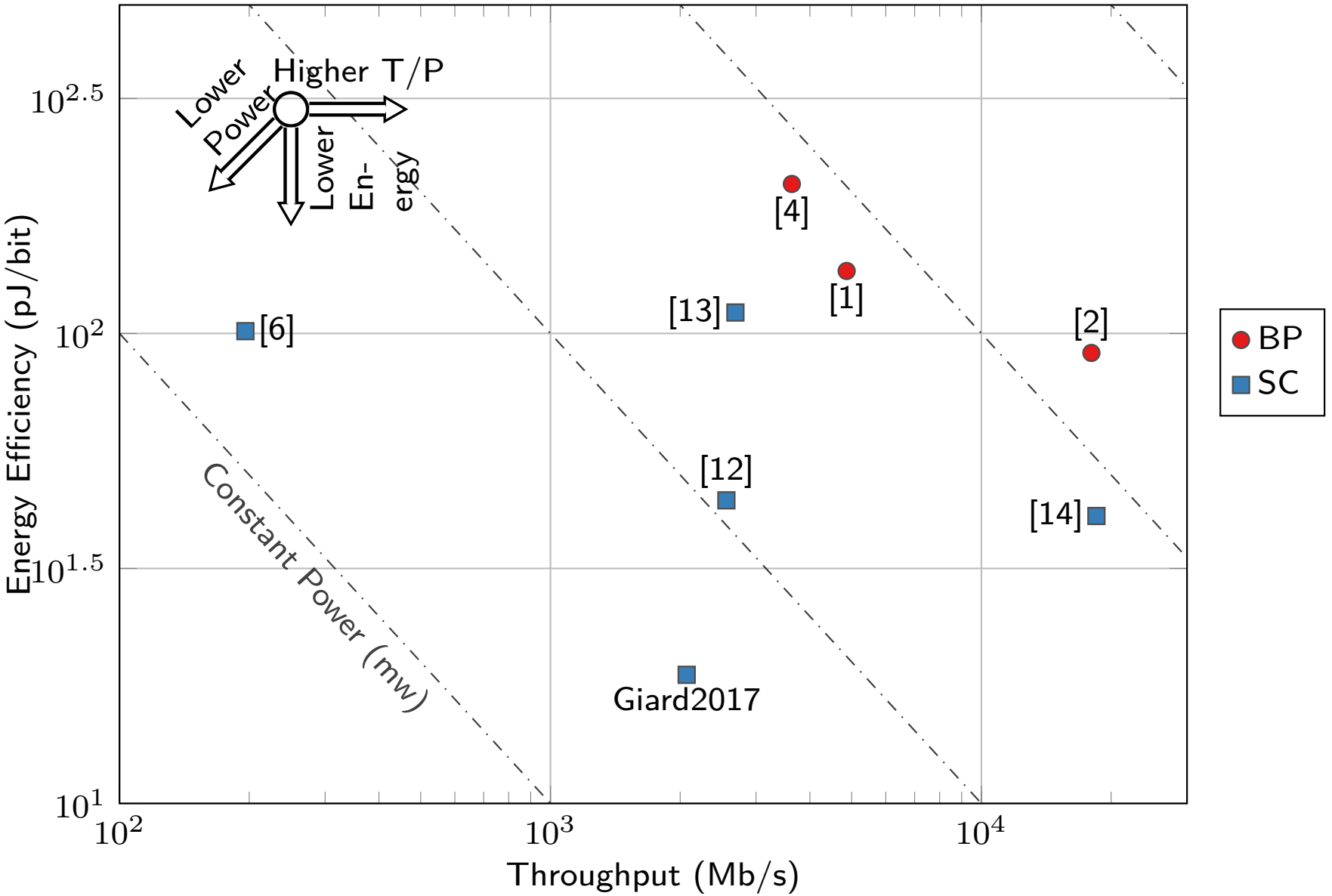
IEEE 802.3an LDPC Decoder Implementations

Work	N	Tech. (nm)	Area (mm^2)	Freq. (MHz)	Schedule	Iter.	T/P (Mb/s)	Voltage (V)	Power (mW)
Cevrero *et al.* [48]	2048	90	5.35	137	Layered	4	11690	1.2	1559
Zhang *et al.* [49]	2048	65	5.35	100	Flooding	8	6670	0.7	144
Zhang *et al.* [49]	2048	65	5.35	700	Flooding	8	47700	1.2	2800
Bao *et al.* [50]	2048	130	18.40	278	Layered	5	9480	1.2	774

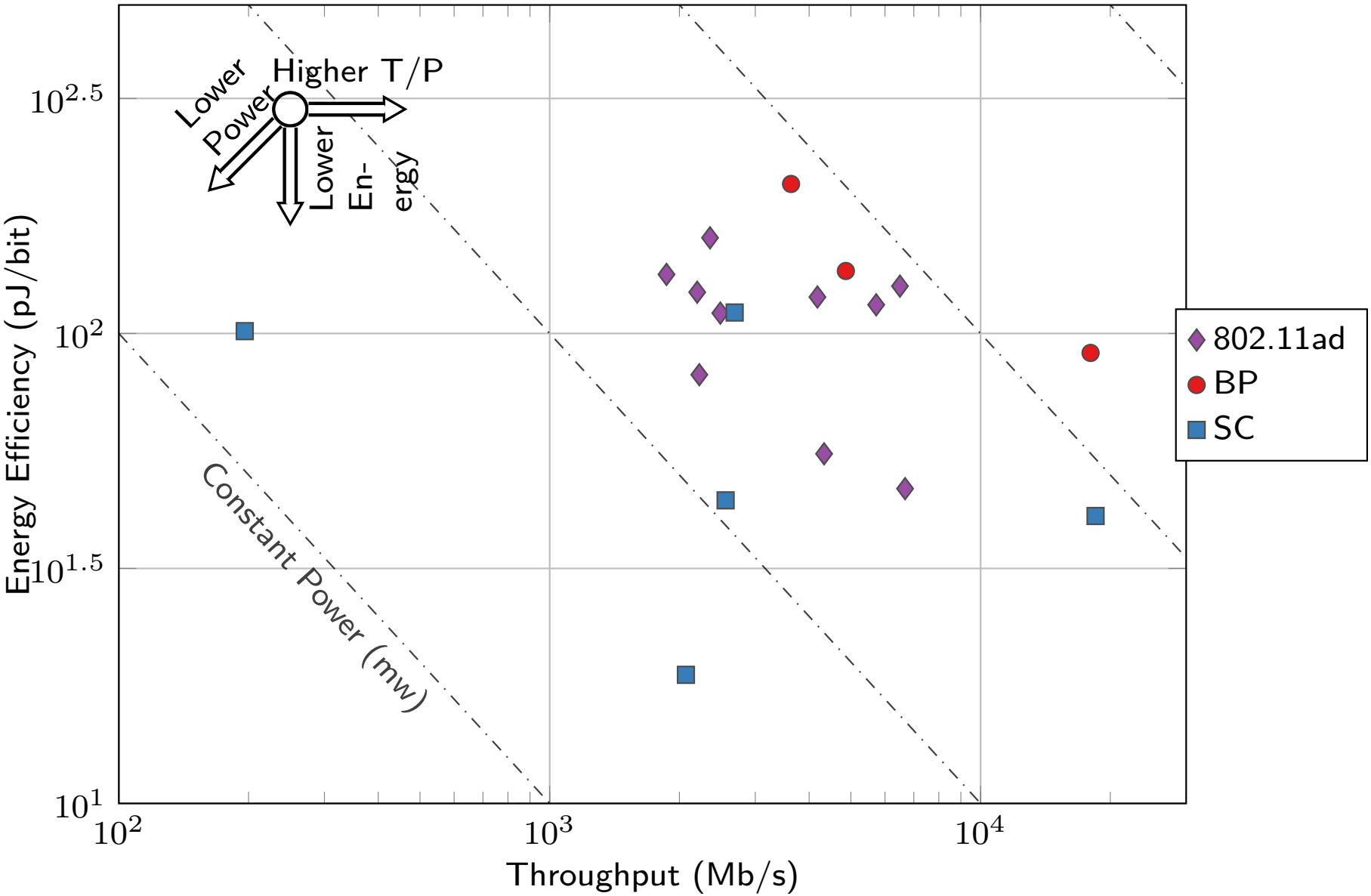
3GPP LTE Turbo Decoder Implementations

Work	K	Tech. (nm)	Area (mm^2)	Freq. (MHz)	Iter.	T/P (Mb/s)	Voltage (V)	Power (mW)
Studer *et al.* [51]	6144	130	3.57	n/a	6	391	1.2	789
Ilnseher *et al.* [52]	6144	65	7.70	450	6	2150	1.1	n/a
Chen *et al.* [53]	6144	65	1.39	512	6	692	1.2	635
Lin *et al.* [54]	6144	40	1.27	252	6	535	0.9	218
Belfanti *et al.* [55]	6144	65	2.49	410	6	1013	1.2	966
Shrestha *et al.* [56]	6144	45	2.43	600	6	1067	0.8	870
Wang *et al.* [57]	6144	90	6.10	625	8	438	1.0	272
Wang *et al.* [57]	6144	90	19.75	625	8	2274	1.0	1450

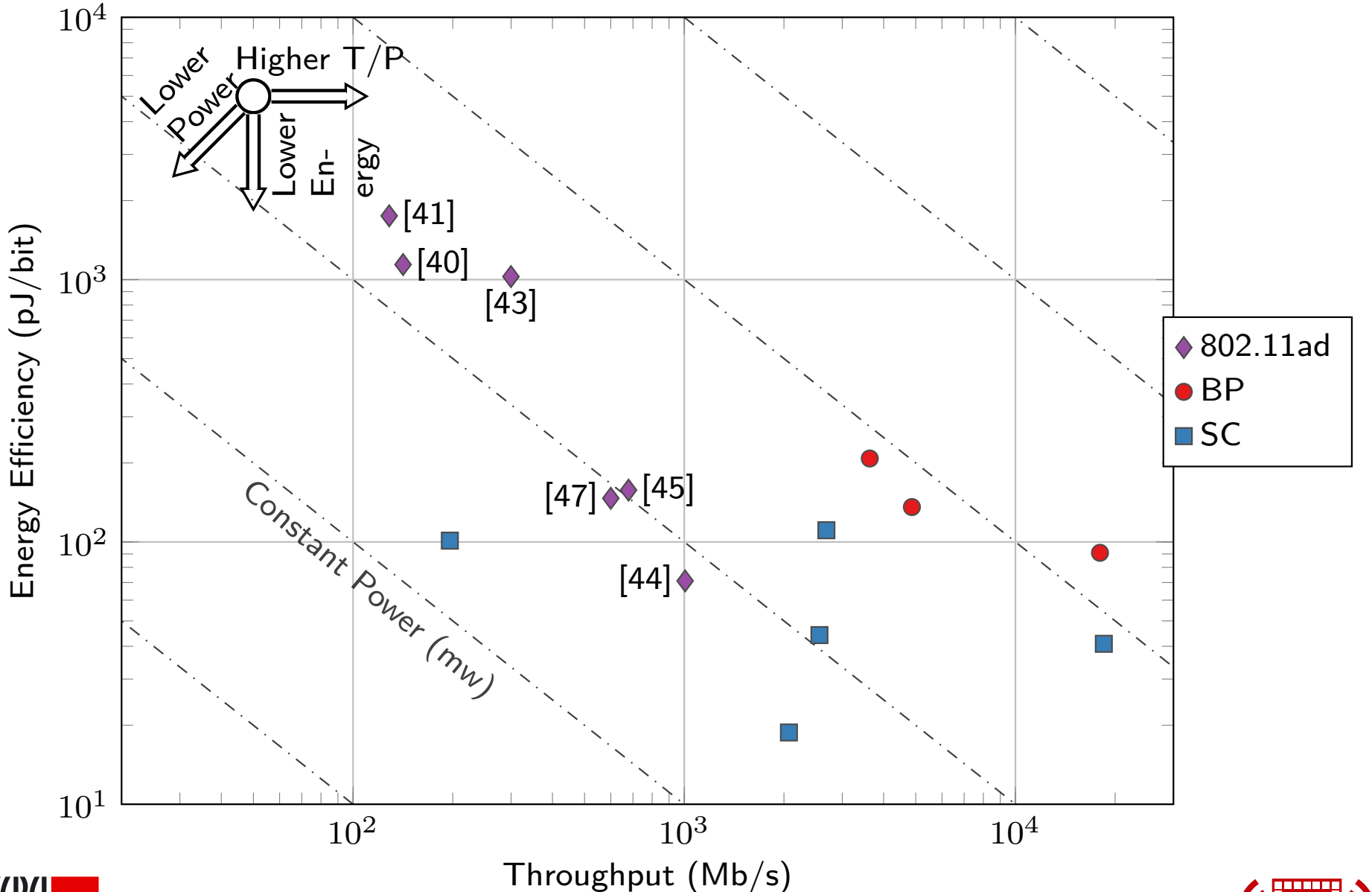
T/P Vs. Energy Eff. for Various Polar Decoders



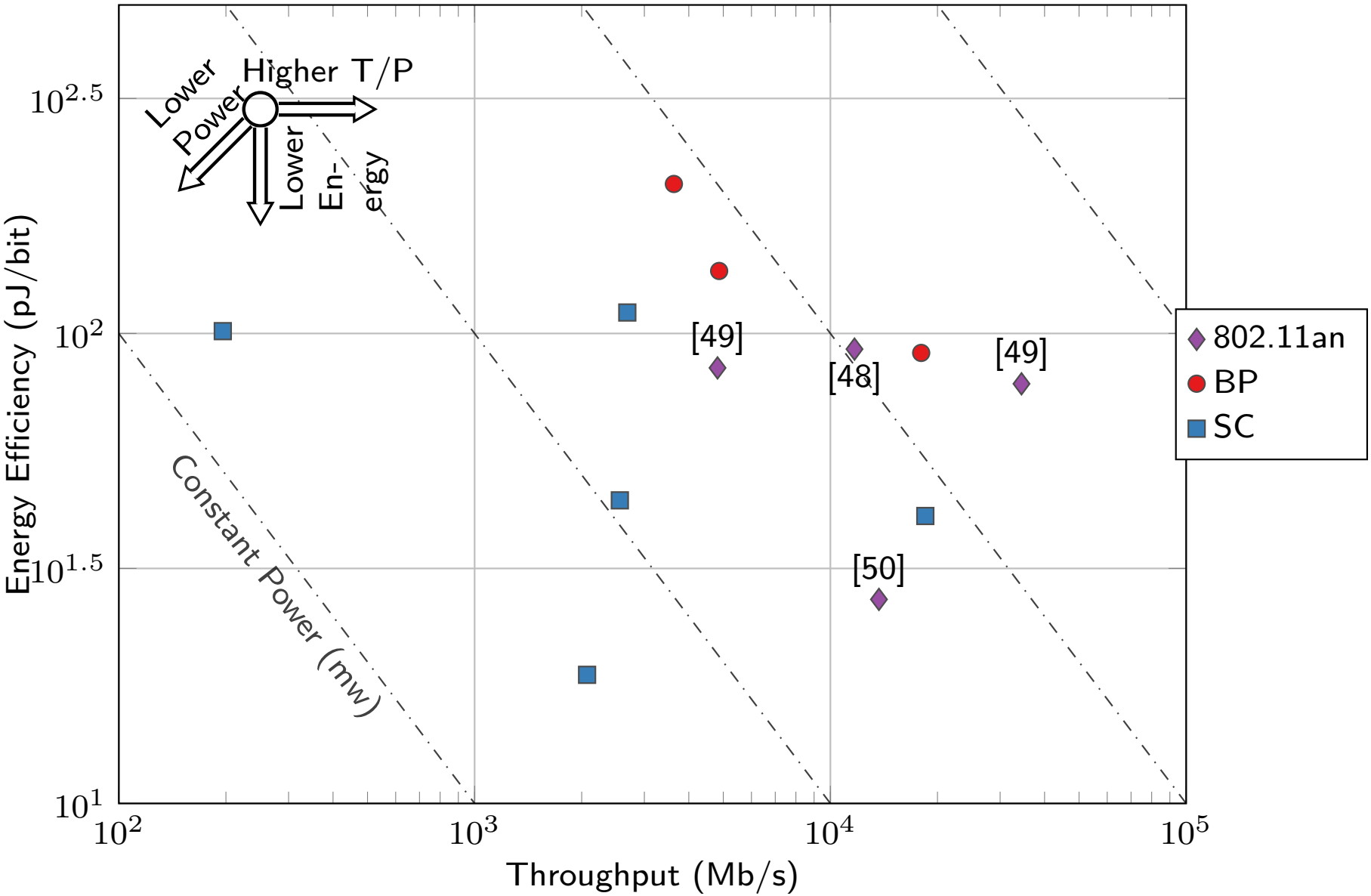
T/P Vs. Energy Eff. – IEEE 802.11ad LDPC



T/P Vs. Energy Eff. – IEEE 802.11n LDPC



T/P Vs. Energy Eff. – IEEE 802.3an LDPC



T/P Vs. Energy Eff. – 3GPP LTE Turbo