UNIVERZITET U BEOGRADU

ELEKTROTEHNIČKI FAKULTET

SRĐAN BRKIĆ

DEKODOVANJE KODOVA SA MALOM GUSTINOM PROVERA PARNOSTI U PRISUSTVU GREŠAKA U LOGIČKIM KOLIMA

doktorska disertacija

Beograd, 2016. godine

Mentor disertacije

dr Predrag N. Ivaniš, vanredni profesor,
Univerzitet u Beogradu, Elektrotehnički fakultet

Članovi komisije

dr Aleksandra Smiljanić, redovni profesor,
Univerzitet u Beogradu, Elektrotehnički fakultet

dr Bane Vasić, redovni profesor,
The University of Arizona, Department of ECE

dr Lazar Saranovac, vanredni profesor
Univerzitet u Beogradu, Elektrotehnički fakultet

dr Goran Đorđević, vanredni profesor,
Univerzitet u Nišu, Elektronski fakultet

Datum odbrane

Sadržaj

Spisak slika

Spisak tabela

Spisak skraćenica

AG	Afina Geometrija
AIG	AND-Inverter Graph
AVS	Aggressive Voltage Scaling
BER	Bit Error Rate
BF	Bit-Flipping
BP	Belief-Propagation
BSC	Binary Symmetric Channel
CMOS	Complementary Metal-Oxide Semiconductor
CN	Check Node
DE	Density Evolution
DRAM	Dynamic Random Access Memory
EG	Euklidska Geometrija
FAID	Finite Alphabet Iterative Decoder
FER	Frame Error Rate
GDBF	Gradient-Descent Bit-Flipping
GF	*Galoa*-ovo polje
HD	Hard-Decision
ITRS	International Technology Roadmap for Semiconductors
GOS	Gate-Output Switching
LDPC	Low Density Parity Check
LER	Line-Edge Roughness
LS	Latin Squares
LUT	Look Up Table
MAE	Message-Aggregation Enhanced
MAJ	MAJority-logic

MP	Message-Passing
MS	Min-Sum
MUDRI	MUltiple-Decoding attempts and Random re-Inicializations
PEG	Progressive Edge Growth
O-S-MAJ	One-Step MAJority logic
PBF	Parallel Bit-Flipping
PDD	Probabilistic Decision Diagram
PEG	Progressive Edge Growth
PG	Progresivna Geometrija
PGDBF	Probabilistic Gradient-Descent Bit-Flipping
PTM	Probabilistic Transfer Matrices
RALF	Reliability Analysis Logic Failures
RDF	Random Dopant Fluctuation
RTL	Register Transfer Level
SBF	Serial Bit-Flipping
SD	Soft-Decision
SPA	Sum-Product Algorithm
SRAM	Static Random Access Memory
TBBF	Two-Bit Bit-Flipping
TPC	Trigonometric Probability Calculation
TK	Taylor-Kuznetsov
TMR	Triple Modular Redundancy
QC	Quasi-Cyclic
VN	Variable Node
VLSI	Very Large-Scale Integration
WBF	Weighted Bit-Flipping
XOR	eXclusive-OR

Spisak publikacija autora

Radovi objavljeni u časopisima

[P1] S. Brkic, P. Ivanis, and B. Vasic, "Reliability of Memories Built from Unreliable Components under Data-Dependent Gate Failures," *IEEE Communications Letters*, vol. 19, iss. 12, pp. 2098–2101, December 2015 (DOI: 10.1109/LCOMM.2015.2496266, ISSN: 1089-7798, IF=1.268) (M22)

[P2] S. Brkic, O.-Al Rasheed, P. Ivanis, and B. Vasic, "On Fault-Tolerance of the Gallager B Decoder under Data-Dependent Gate Failures," *IEEE Communications Letters*, vol. 19, iss. 8, pp. 1299–1302, Avgust 2015 (DOI: 10.1109/LCOMM.2015.2442981, ISSN: 1089-7798, IF=1.268) (M22)

[P3] P. Ivanis, D. Drajic, and S. Brkic, "Cross-Layer Combining of Adaptive Modulation and Truncated ARQ in Multichannel Beamforming MIMO Systems," *Radioengineering*, vol. 24, no. 4, pp. 1050-1059, December 2015 (DOI: 10.13164/re.2015.1050, ISSN: 1210-2512, IF=0.796) (M23)

[P4] S. Brkic, P. Ivaniš, "Energy detector performance in Rician fading channel," *Serbian Journal of Electrical Engineering*, vol. 10, no. 1, pp. 37-46, February 2013. (M51)

[P5] O.-Al Rasheed, S. Brkic, P. Ivanis, B. Vasic, "Performance Analysis of Faulty Gallager-B Decoding of QC-LDPC Codes with Applications," *Telfor Journal*, Vol 6, No 1, pp. 7-11, November 2014. ISSN 1821-3251. (M53)

[P6] S. Brkic, P. Ivanis, Goran Đorđević, Bane Vasic,"Symbolic Analysis of Faulty Logic Circuits under Correlated Data-Dependent Gate Failures," *Telfor Journal*, Vol 6, No 1, pp. 2-6, November 2014. ISSN 1821-3251. (M53)

[P7] S. Brkic, P. Ivanis, "Performance Evaluaton of HARQ Technique with UMTS Turbo Code, *Telfor Journal*, Vol 3, No 2, pp. 86-89, November 2011. ISSN 1821-3251. (M53)

Radovi prezentovani na konferencijama

[P8] B. Vasic, P. Ivanis, S. Brkic, "Low complexity memory architectures based on LDPC codes: benefits and disadvantages," *In Proc. of 12th International Conference on Advanced Technologies Systems and Services in Telecommunications (TELSIKS 2015)*, Nis, Serbia, Oct. 2015. (M31)

[P9] S. Brkic, P. Ivanis, G. Djordjevic, B. Vasic, "Taylor-Kuznetsov fault-tolerant memories: a survey and results under correlated gate failures," in *Proc. IEEE TELSIKS 2013*, Nis, Serbia, October 16th-19th, 2013, pp. 455-462 ISBN: 978-1-4799-0900-1. (M31)

[P10] P. Ivaniš, V. Blagojević, M. Stojnić, S. Brkić, "User Cooperation Diversity in Cognitive Radio Systems," in *Proc. SAUM 2012*, Niš, Serbia, November 14th-16th, 2012, pp. 72-79 (ISBN 978-86-6125-072-9) (M31)

[P11] S. Brkic, P. Ivanis, and B. Vasic, "Guaranteed Error Correction of Faulty Bit-Flipping Decoders under Data-Dependent Gate Failures," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 2016)*, Barcelona, Spain, July 10-15 (M33)

[P12] S. Brkić, B. Vasić, P. Ivaniš, David Declercq, "Message-Aggregation-Enhanced Iterative Hard-Decision Decoders, *Information Theory and Applications Workshop (ITA)*, San Diego, USA, January 31 - February 5 2016. (M33)

[P13] B. Vasić, P. Ivaniš, S. Brkić, V. Ravanmehr "Fault-Resilient Decoders and Memories made of Unreliable Components," in *Proc. Information Theory and Applications Workshop (ITA 2015)*, San Diego, USA, February 1-6 2015. (M33)

[P14] S. Brkic, P. Ivanis, Bane Vasic, "Analysis of one-step majority logic decoding under correlated data-dependent gate failures," in *Proc. IEEE International Symposium on Information Theory (ISIT 2014)*, pp. 2599 - 2603, Honolulu, USA, June 29-July 4 2014. (M33)

[P15] S. Brkic, P. Ivanis, G. Djordjevic, B. Vasic, "Symbolic analysis of faulty logic circuits in the presence of correlated gate failures," in *Proc. IEEE TELFOR 2013*, Belgrade, Serbia, November 26th-28th, 2013, pp. 369-372. ISBN: 978-1-4799-1419-7. (M33)

[P16] O.-Al Rasheed, S. Brkic, P. Ivanis, B. Vasic, "Performance Analysis of Faulty Gallager-B Decoding of QC-LDPC Codes," in *Proc. IEEE TELFOR 2013*, Belgrade, Serbia, November 26th-28th, 2013, pp. 323-326. ISBN: 978-1-4799-1419-7. (M33)

[P17] S. Brkic, M. Eric, "FPGA Implementation of Joint Spatio-Temporal Spectrum Sensing Algorithm Based on Direct Localization Method," in *Proc. IEEE TELSIKS 2013*, Nis, Serbia, October 2013, pp. 301- 304. ISBN: 978-1-4799-0900-1. (M33)

[P18] S. Brkić, P. Ivaniš, "Joint Optimization of Adaptive Modulation and Eigenchannel Power Allocation in Dual-Branch SVD-MIMO Systems," in *Proc. SAUM 2012*, Niš, Serbia, November 14th-16th, 2012, pp. 343-346. (ISBN 978-86-6125-072-9), (M33)

[P19] S. Brkic, P. Ivanis, "Performances of HARQ Technique with UMTS Turbo Code in Nakagami Fading Channels," in *Proc. IEEE TELSIKS 2011*, Nis, Serbia, October 5th-8th, 2011, pp. 459-462. ISBN: 978-1-4577-2018-5. (M33)

[P20] S. Brkić, D. Lazarević, M. Simić, "Procena performansi detektora signala zasnovanog na sopstvenim vrednostima kovarijacione matrice," *XIII Međunarodni naučno-stručni Simpozijum Infoteh-Jahorina*, vol. 13, str. 361-365, Mart 2014. (M63)

[P21] P. Ivaniš, S. Brkić, G. Đorđević, B. Vasić, "Savremene tehnike za projektovanje pouzdanih čipova napravljenih od nepouzdanih komponenata," *Zbornik XXXII Simpozijuma o novim tehnologijama u poštanskom i telekomunikacionom saobraćaju (POSTEL 2014)*, Beograd, 2-3. decembra 2013, str. 277-286. ISBN 978-86-7395-328-1. (M63)

[P22] P. Ivaniš, M. Erić, S. Brkić, M. Janjić, "Tehnike za efikasno korišćenje spektra: prikaz nekih rezultata istraživanja," *Zbornik XXXI Simpozijuma o novim tehnologijama u poštanskom i telekomunikacionom saobraćaju (POSTEL 2013)*, Beograd, 3-4. decembra 2013, str. 233-242. ISBN 978-86-7395-314-4. (M63)

[P23] S. Brkić, P. Ivaniš, "Performanse kooperativnog spectrum sensinga u kanalu sa generalizovanim K fedingom," *Zbornik 57. konferencije ETRAN*, Zlatibor, 3-6. juna 2013, str. TE2.2.1-6. (M63)

[P24] S. Brkić, D. Lazarević, P. Ivaniš, "FPGA implementacija sum-product algoritma za dekodovanje LDPC kodova," *INFOTEH JAHORINA 2013*, Vol 12, Ref. KST-3-2, Istočno Sarajevo, 20-22. Mart 2013, str. 428-433. ISBN 978-99955-763-1-8. (M63)

[P25] S. Brkić, P. Ivaniš, "Performanse detektora energije u kanalu sa Rajsovim fedingom," *ETRAN 2012*, Zlatibor, Srbija, 11-14. Jun 2012, TE 3.5-1-4. ISBN 978-86-80509-67-9. (M63)

[P26] S. Brkić, D. El Mezeni, L. Saranovac, J. Popović Božović, M. Erić, "Evaluacija razvojnih platformi za sisteme spectrum sensing-a," *Zbornik radova INFOTEH-JAHORINA 2012*, pp. 401-405, Vol. 11, mart 2012. (M63)

[P27] S. Brkić, D. El Mezeni, L. Saranovac, J. Popović Božović, "FPGA dizajn kanalizatora spektra na bazi polifazne banke filtara," *Zbornik radova TELFOR 2011*, Beograd, Srbija, novembar 2011. (M63)

[P28] S. Brkić, P. Ivaniš, "Procena performansi hibridne ARQ tehnike sa UMTS turbo kodom," *XVIII Telekomunikacioni forum TELFOR 2010*, Beograd, 23-25. Novembar 2010, str 521-524. ISBN 978-86-7466-392-9. (M63)

[P29] S. Brkić, "Simulaciona analiza kodova za cikličnu proveru redundanse," *XVII Telekomunikacioni forum TELFOR 2009*, Beograd, 24-26. Novembar 2009, str 1439-1442. ISBN 978-86-7466-392-9. (M63)

Rezime

Sve veća integracija poluprovodničkih tehnologija, varijacije nastale usled nesavršenosti procesa proizvodnje, kao zahtevi za smanjenjem napona napajanja čine elektronske uređaje inherentno nepouzdanim. Agresivno skaliranje napona smanjuje otpornost na šum i dovodi do nepouzdanog rada uređaja. Široko je prihvaćena paradigma prema kojoj se naredne generacije digitalnih elektronskih uređaja moraju opremiti logikom za korekciju hardverskih grešaka.

Jedina za sada poznata klasa linearnih blok kodova koja je otporna na otkaze logičkih komponenti su kodovi sa malom gustinom provera parnosti (eng. *Low-Density Parity-Check*, LDPC). Atraktivnost LDPC kodova ogleda se u teorijskoj garanciji da se hardverska redundansa potrebna za obavljanje pouzdane operacije povećava samo linearno sa dužinom koda, čak i kada su logička kola nepouzdana. Poznato je da visok nivo pouzdanosti obezbeđuju *message-passing* i *bit-flipping* dekoderi LDPC kodova, koji donose tvrde odluke, i tako smanjuju propagaciju grešaka izazvanu otkazima logičkih kola, što nije slučaj kod složenijih dekodera koji donose meke odluke.

Ovaj rad pokušava da odgovori na neka fundamentalna pitanja kao što su: *kakav je uticaj otkaza logičkih kola na performanse dekodera LDPC kodova* i *da li je moguće garantovati ispravljanje određenog broja greška u kanalu ako se dekodovanje obavlja nepouzdanim logičkim kolima*. Odgovor na prvo pitanje nije trivijalan u svetlu novih otkrića koja preispituju intuitivan zaključak da nepouzdanost posledično dovodi do negativnih efekata. U ovom radu pokazano je da vremenski korelisani otkazi logičkih kola mogu delovati pozitivno na proces iterativnog dekodovanja *Gallager B* dekoderom kvazi-cikličnih LDPC kodova i smanjiti nivo zaostale greške nakon dekodovanja. Kod drugih grupa kodova nije primećen dobitak uzrokovan hardverskom nepouzdanošću, ali je predložena modifikacija dekodera koja obezbeđuje visok nivo otpornosti na otkaze logičkih elemenata. Tolerantnost *Gallager B* dekodera iskorišćena je pri dizajniranju novog *hibridnog dekodera*, koji objedinjuje *bit-flipping* i *message-passing* principe u jedinstveno rešenje niske kompleksnosti. Kompatibilnost nepouzdanog *Gallager B*

dekodera sa inovativnim *bit-flipping* pristupom, u kome varijabilni čvorovi dekodera komuniciraju direktno, dovodi do performansi koje prevazilaze većinu poznatih rešenja slične ili veće kompleksnosti.

U nedostatku eksplicitnih metoda za konstrukciju LDPC kodova koji ispravljaju određeni broj grešaka, analiza garantovane korektivne sposobnosti različitih LDPC kodova je jedan od najznačajnijih praktičnih problema iz ove oblasti. Do sada je pružen rigorozan matematički dokaz da svega nekoliko iterativnih dekodera ima sposobnost ispravljanja fiksne frakcije grešaka i to pod pretpostavkom pouzdanih logičkih operacija. U ovom radu je pokazano da podgrupa LDPC kodova nazvana ekspander kodovi, dekododvana jednostavnim *bit-flipping* dekoderom, može ispraviti fiksnu frakciju grešaka nastalih u toku prenosa kroz telekomunikacioni kanal, čak i kada su logička kola dekodera podložna vremenski korelisanim otkazima. Dodatno, numerički su određene gornja i donja granica korektivne sposobnosti dekodera, za različite parametre LDPC kodova.

Još od pionirskih radova *Taylor*-a i *Kuznetsov*-a poznato je da memorije kodovane LDPC kodovima postižu teorijski optimalnu pouzdanost memorisanja informacija. Njihova nešto veća kompleksnost nego rešenja koja uključuju *Hamming*-ove, BCH ili *Reed-Solomon*-ove kodove dugo su predstavljala prepreku njihove praktične primene. Međutim, sa povećanjem nepouzdanosti komponenata memorije, pitanje je vremena kada postojeća rešenja neće moći da pruže željenu zaštitu memorisanih podataka. Kako se očekuje se da će u decenijama koje slede memorije bazirane na LDPC kodovima biti značajnije zastupljene, ovaj rad doprinosi i u toj oblasti. Razmatrana memorijska arhitektura bazirana na *bit-flipping* algoritmu pokazala je visok stepen robusnosti na vremenski korelisane otkaze komponenti. Teorijska razmatranja dopunjena su i sa numerički određenim granicama broja otkaza koje je moguće tolerisati.

Abstract

Due to huge density integration increase, lower supply voltages, and variations in technological process, complementary metal-oxide-semiconductor (CMOS) and emerging nanoelectronic devices are inherently unreliable. Moreover, the demands for energy efficiency require reduction of energy consumption by several orders of magnitude, which can be done only by aggressive supply voltage scaling. Consequently, the signal levels are much lower and closer to the noise level, which reduces the component noise immunity and leads to unreliable behavior. It is widely accepted that future generations of circuits and systems must be designed to deal with unreliable components.

Recently, there has been a surge in interest in error control schemes that can ensure fault-tolerance in unreliable hardware. The only known class of codes resilient to logic gate faults are low-density parity-check (LDPC) codes. Their attractiveness lays in the theoretical guarantee that the decoding hardware overhead required to ensure reliable operation grows only linearly with the code length even when logic gates are faulty. Such fault tolerant decoders are based on message-passing and bit-flipping decoding algorithms, which unlike more complex algorithms, limit the error propagation in a decoder caused by faulty logic gates.

The aim of the thesis is to answer to several fundamental questions as: *what effect logic gate failures have to performance of decoders of LDPC codes* and *is it possible to guarantee correction of channel errors if unreliable logic gates are used in the decoding process*. The answer to the first question is not trivial in the light of new discoveries which reassess the intuitive conclusion that unreliability always leads to performance degradation. In this theses we show that data-dependent gate failures can have positive impact on Gallager B decoding algorithm applied on quasi-cyclic LDPC codes, and reduce the residual error level. The other classes of LDPC codes do not gain the logic gate failures, but we present the decoder modification that ensures high level of gate failure robustness. This fact is used to design a new *hybrid decoder*, which unites bit-flipping and message-passing principles into a single low complexity

solution. Compatibility of faulty Gallager B decoder with innovative bit-flipping principle, in which variable nodes communicate directly, leads to superior performance compared to other solutions with similar or higher complexity.

Lack of constructive approaches which guarantee correction of a certain number of channel errors makes error correction analysis of LDPC codes one of the most significant open research problems. Until present day strict mathematical proof for correction of a fixed fraction of errors was provided only for a few practically significant decoders. In this thesis we show that a subclass of LDPC codes, named expander codes, decoded by the faulty bit-flipping decoder, can correct a fixed fraction of channel errors. In addition, numerically obtained, upper and lower bounds on the number correctable error patterns are provided.

Since pioneering work of Taylor and Kuznetsov it is known that LDPC-enhanced memories perform optimally in asymptotic case. However, their higher complexity compared to other solutions that include Hamming, BCH or Reed-Solomon codes was the main obstacle for their practical implementations. On the other hand, as hardware unreliability constantly increases, it is expected that current state-of-the-art solutions will not be able to provide the desired level of memory reliability, and the LDPC-based memories will be more exploited in years to come. This thesis contributes also in the area of reliable storage. It is shown that proposed memory architecture, based on the bit-flipping decoder, can tolerate a fraction of component failures when gate failures are data-dependent and correlated in time. Theoretical results are enriched by numerically obtained upper bounds on the fraction of tolerable component failures.

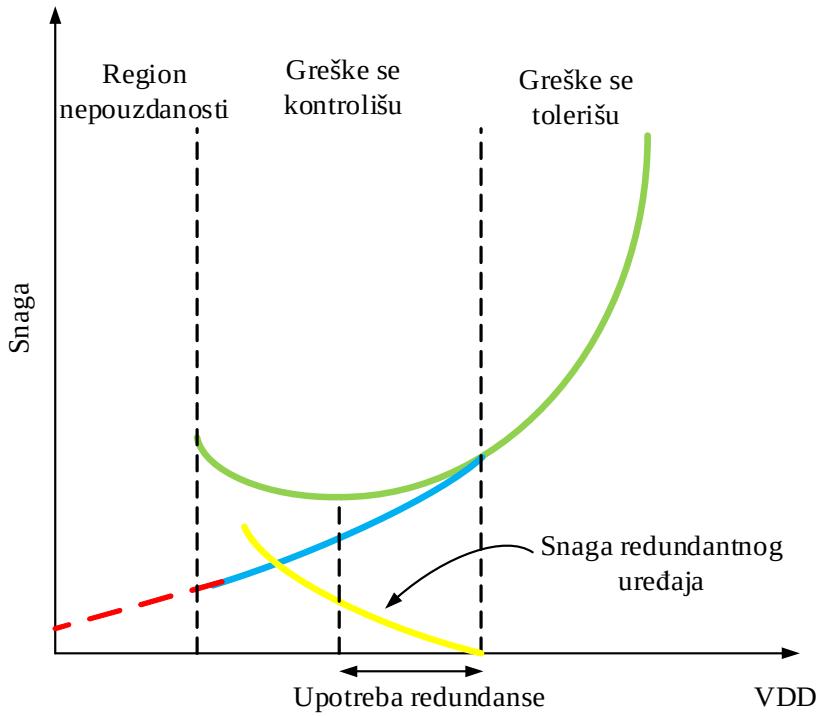
Poglavlje 1

Uvod

U toku proteklih nekoliko dekada nezaustavljiv napredak poluprovodničkih tehnologija doveo je do veoma malih, brzih i efikasnih čipova. Kako se tražnja za energetski efikasnim (zelenim) tehnologijama nastavlja, čitav niz nano-tehnologija su istraživane u kontekstu procesiranja ili skladištenja informacija. Iako je teško proceniti koje će konkretno tehnike biti baza za dalji razvoj računarstva, prepoznato je da će usled skaliranja (smanjenja) elektronskih komponenti i nesavršenosti procesa proizvodnje, one biti inherentno nepouzdane. Čak je i u tradicionalnim poluprovodničkim tehnologijama, kao na primer CMOS (eng. *Complementary Metal–Oxide Semiconductor*) primetno narušavanje pouzdanosti, tako da preovladava mišljenje da su otkazi tranzisotra glavna prepreka daljem napretku ove oblasti. Značaj problema nepouzdanosti prepoznat je i u ITRS (eng. *International Technology Roadmap for Semiconductors*) dokumentu iz 2010. godine [6], gde je navedeno da je obezbeđivanja robusnosti na otkaze tranzistora jedan od pet najznačajnijih izazova novih poluprovodničkih tehnologija. Međutim, kako se navodi u izveštajima Agencije za standardizaciju i tehnologije Sjedinjenih Američkih Država [7], i samo razumevanje faktora koji utiču na performanse i pouzdanost MOS tehnologija manjih od 30-nm je samo po sebi izazovno. Iako nekoliko značajnih kompanija, među kojima su i IBM i Intel, aktivno učestvuju u istraživanjima iz ove oblasti, ne nazire se tehnološko rešenje koje bi obezbedilo pouzdanost novih elektronskih komponenata. Zbog toga se prihvata nova paradigma razvoja poluprovodničkih tehnologija prema kojoj se komponente proizvode inherentno nepouzdane, a njihov pouzdan rad se obezbeđuje isključivo na funkcionalnom nivou, koji često podrazumeva dodavanje redundantnih komponenti u uređaj [8]. Promena paradigme izmešta problem dizajniranja elektronskih sistema iz domena elektronike u užem smislu, pa se on sve više posmatra kao interdisciplinarni problem. Od posebnog interesa su rešenja koja

pruža teorija informacija, kao naučna disciplina koja proučava pouzdanost prenosa informacija kroz nepouzdane medijume.

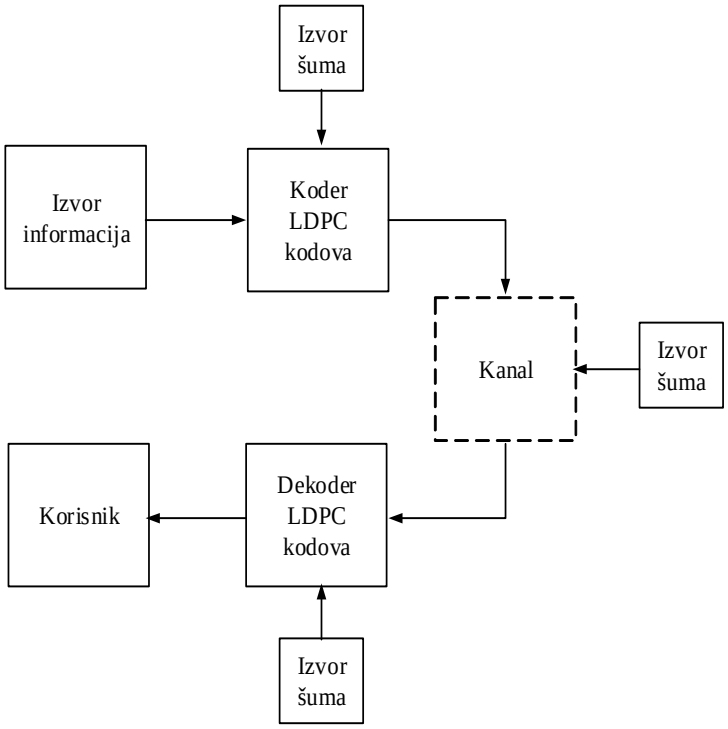
Problem nepouzdanosti ilustrovan je na slici 1.1, gde je opisana zavisnost snage koju troši elektronski uređaj od napona napajnja tranzistora (VDD). Umereno smanjivanje napona napajanja dovodi do smanjenja disipacije snage uređaja, pri čemu se greške koje nastaju kao posledica tog smanjenja mogu tolerisati. Međutim, sa daljim smanjenjem nivoa napajanja greške postaju sve učestalije i potrebno ih je kontrolisati. Dodavanje redundanse u uređaj povećava izračenu snagu, pri čemu se teži ka rešenju koje će obezbediti da dodatno smanjenje napona napajanja kompenzuje energiju redundantnog uređaja. Redundantni uređaj treba da obezbedi željenu pouzdanost operacija koju obavlja uređaj, pri čemu je i on podložan greškama, jer se napaja smanjenim naponom. Ako se želi zadržati nivo pouzdanosti uz dodatno smanjenje napona, potrebno je usložniti redundantni uređaj, što može do proizvesti veću snagu od one koja bi bila potrebna u slučaju da se redundantni uređaj ne koristi, a napon napajanja ne smanjuje. Jedan od istraživačkih zadataka, je optimizovanje redundantnog uređajnja tako da se region njegove upotrebne maksimizuje. Pri ekstrebno malim nivoima napajanja greške nije moguće korigovati, pa uređaj nije moguće ni koristiti.



Slika 1.1: Ilustracija problema nepouzdanosti kao posledica smanjenja napona napajanja tranzistora.

Izbor optimalnog redundantnog uređaja pripada oblasti teorije informacija, koja u protekle dve decenije prolazi kroz renesansni period. Osnove teorije informacija postavio je *Claude Shannon* u svom fundamentalnom radu iz 1948. godine [9], koji je definisao i danas važeće granice pouzdanog komuniciranja. Međutim, optimalan način dodavanja redundanse u formi zaštitnog koda predstavlja otvoren problem. Tokom godina su konstruisani različiti zaštitni kodovi čije su performanse bile daleko od optimalnih. Neki od praktično najznačajnih su *Hamming*-ovi, BCH ili *Reed-Solomon*-ovi kodovi. Ozbiljan proboj u oblasti zaštitnih kodova napravljen je tek devedesetih godina prošlog veka sa otkrivanjem dve konkurentske klase linearnih blok kodova koje su približile veoma blizu *Shannon*-ove granice – *turbo kodova* [10] i kodova sa malom gustinom provera parnosti (eng. *Low-Density Parity-Check*, LDPC) [11]. Manja kompleksnost, veća raznovrsnost u izboru kodnih količnika, kao i algoritama dekodovanja favorizovala je LDPC kodove, koji su se brzo izdvojili kao primarni interes istraživačke zajednice. Može se sa sigurnošću tvrditi da su LDPC kodovi najznačajnija i najistraženija klasa zaštitnih kodova i kao takvi su našli primenu u nekoliko značajnih telekomunikacionih protokola, kao što su DVB-S2, IEEE 802.3an – 10GBASE-T, IEEE 802.11n (WiFi) i IEEE 802.16e (WiMax).

U klasičnoj teoriji zaštitnog kodovanja informacija koja se prenosi koduje se zaštitnim kodom na predaji, da bi se nakon prolaska kroz kanal na prijemoj strani dekodovala. Osnovna pretpostavka klasičnog *Shannon*-ovog rada je da nepouzdanost u sistemu prenosa potiče od medijuma za prenos, dok se operacije kodovanja i dekodovanja obavljaju deterministički. Međutim, hardverska nepouzdanost koder i dekoder pretvara u stohastičke komponente, kako je to prikazano na slici 1.2. Nova paradigma zaštitnog kodovanja ruši postojeće granice uspešnog komuniciranja, pri čemu nove granice još uvek nisu poznate. Na problem uspešnog komuniciranja, u kontekstu memorisanja informacija, pažnju su skrenuli Vasić i *Chilappagari* u sada već klasičnom članku iz 2007. godine [12], nakon čega se ovoj oblasti pridaje sve veći značaj. Dominantan pravac istraživanja je sagledavanje akumuliranog znanja klasnične teorije kodovanja, pre svega teorije LDPC kodova, u novom kontekstu pouzdnosti sistema sastavljenih od nepouzdanih komponenti. Dizajniranje dekodera koji imaju sposobnost ispravljanja grešaka nastalih u toku prenosa kroz kanala, a da pri tome i sami unose greške, ili konstruisanje memorijskih arhitektura koje iako napravljanje od nepouzdanih komponenti ipak uspevaju da uspešno čuvaju informacije, samo su neki od izazovnih problema koji se rešavaju u ovoj uzbudljivoj oblasti istraživanja.

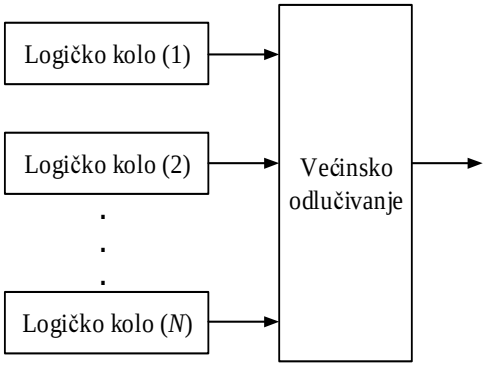


Slika 1.2: Nova paradigma razvoja teorije zaštitnog kodovanja.

1.1 Pouzdano računanje i memorisanje informacija

Problem pouzdanog računanja elektronskim uređajima sastavljenim od nepouzdanih komponenti formulisao je čuveni matematičar *von Neumann* u pionirskom članku iz 1956. godine [13]. Sve komponente elektronskog uređaja (ili automata kako se uređaj naziva u *von Neumman*-ovoj terminologiji) su nepouzdane, pri čemu se njihova nepouzdanost modeluje probabilistički – svaka komponenta otkazuje sa fiksnom unapred poznatom verovatnoćom. Da bi povećao pouzdanost automata, *von Neumann* je predložio da se svaka komponenta multipleksira N puta, pri čemu se finalna odluka donosi prema pravilu većinskog odlučivanja. *Von Neumann*-ovo multipleksiranje logičkih kola ilustrovano je na slici 1.3.

Specijalan slučaj multipleksiranja kada je $N = 3$ naziva se TMR (eng. *Triple Modular Redundancy*) i, zbog svoje jednostavnosti, zauzima značajno mesto u mnogim praktičnim realizacijama [14–16]. *Von Neumann*-ova šema odgovara kodu sa ponavljanjem u klasičnoj teoriji informacija, za koji se zna da zahteva veliku redundansu, ako je nepouzdanost komponenata velika (broj komponenata mora da se bar utrostruči). Dodatno, *von Neumann*-ov rad nije u duhu *Shannon*-ovog zapažanja da je proizvoljno pouzdan prenos informacija moguće postići dodavanjem redundanse koja se samo linearno povećava (kompleksnot $O(k)$) sa količinom informacija koje je potrebno preneti (k). S druge strane, *von Neumann* je primetio da se

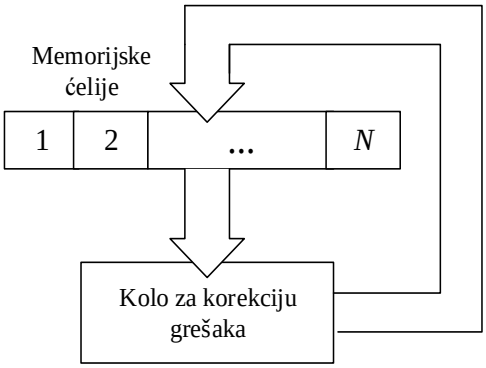


Slika 1.3: Blok šema *von Neumann*-ovog multipleksiranja.

asimptotski pouzdano računanje neke *Boole*-ove funkcije od k komponenata postiže šemom čija kompleksnost iznosi $O(k \log k)$. Formalni dokaz *von Neumann*-ove tvrdnje pružio je *Taylor* [17]. Kasniji pokušaji da se konstruiše šema pouzdanog računanja *Boole*-ovih funkcija, čija bi kompleksnost iznosila $O(k)$ bili su, najblaže rečeno, delimično uspešni. *Elias* [18] je prvi predložio da se linerani blok kodovi iskoriste za povećanje pouzdanosti *Boole*-ovih funkcija. Međutim, osim za neke partikularne slučajeve, kao što su XOR funkcije, nije uspeo da konstruiše kodovanu šemu koja bi prevazišla *von Neumann*-ovo multipleksiranje. *Dobrushin* i *Ortyukov* [19] su kasnije i matematički dokazali postojanje funkcija za koje to ni teorijski nije moguće uraditi. Do sličnih zapažanja nešto ranije u odvojenoj studiji došli su *Winograd* i *Cowan* [20]. Drugim rečima, pomenuti radovi su pokazali da odnos broja pouzdanih komponenti proizvoljno izabrane *Boole*-ove i broja nepouzdanih komponenti potrebnih za njen pouzdan rad ne može biti konstantan čak ni u asimptotskom slučaju. Ovaj odnos se često naziva *računarskim kapacitetom* (eng. *computational capacity*).

U svom kritičkom osvrtu na *von Neumann*-ov rad, *Pippenger* [21] je primetio da je *von Neumann*-ov uslov da sve komponente uređaja moraju biti nepouzdane suviše restriktivan i da "kritične tačke" uređaja moraju biti pouzdane. Ilustraciju navedene tvrdnje moguće je pronaći u *Taylor*-ovom radu posvećenom pouzdanom skladištenja informacija iz 1968. godine [22]. *Taylor* je predložio šemu u kojoj bi se informacija čuvala u memorijskim ćelijama u formi kodne reči nekog LDPC koda. Usled nepouzdanosti memorijskih elemenata, ćelije se periodično ažuriraju korišćenjem kola za korekciju grešaka, kako je to ilustrovano na slici 1.4. Izuzetnost *Taylor*-ovog rada ogleda se u činjenici da je uspeo da dokaže da memorija može pouzdano skladištiti informacije proizvoljno dug vremenski interval, čak i kada su sve komponente nepouzdane. Deo koji mora biti savršeno pouzdan je kolo na osnovu koga se vrši ekstrakcija (dekodovanje) informacija iz memorije, koje se uključuje onda kada treba pročitati

skladištene informacije. *Taylor*-ov koncept proširio je *Kuznetsov* [23], pa se često u literaturi memorijska arhitektura koju su predložili naziva TK (*Taylor-Kuznetsov*) memorijom. *Taylor* je takođe pokazao da kompleksnost kola za korekciju grešaka (LDPC dekodera) raste samo linearno sa brojem memorijskih ćelija koje čuvaju informaciju, pa je u ovom slučaju moguće postići nenulti računarski kapacitet, koji se u kontekstu memorija još naziva i *memorijskim kapacitetom* (eng. *storage capacity*). U kasnijim radovima *Chilappagari* i Vasić [24] i *Varshney* [25] predložili su memorijske arhitekture koje imaju sposobnost pouzdanog čuvanja informacija, a njihova kompleksnost je značajno niža od originalne TK memorije.



Slika 1.4: Ilustracija *Taylor*-ove kodovane memorije.

1.2 Značaj kodova sa malom gustinom provera parnosti za pouzdanost sistema

LDPC kodove predložio je *Gallager* 1963. godine [26], ali su bili zaboraljeni dok na njih nije ukazao *MacKay* krajem prethodnog veka [11], koji je pokazao da postižu performanse uporedive sa ranije otkrivenim turbo kodovima. *Kschischang* [27] je primetio da se LDPC kodovi mogu dekodovati propagirajući poruke preko grafa, što je jedan od ključnih rezultata koji je doprineo njihovoj popularnosti. On je uspostavio vezu između iterativnih dekodera LDPC kodova i različitih grafičkih modela i algoritama već razvijenih u teoriji ekspertskih sistema [28, 29]. Značajan doprinos razvoju teorije LDPC kodova dali su i *Wiberg* [30] i *Forney* [31], kao i *Tanner* koji je predložio grafovski opis generalizovne verzije LDPC kodova. Konačno, *Richardson* i *Urbanke* [5, 32] su pokazali da pored izuzetnih praktičnih karakteristika iregularni LDPC kodovi dostižu i *Shannon*-ov kapacitet, čime su i teorijski verifikovani kao nezamenljiv produkt teorije zaštitinih kodova.

Novi pravac istraživanja LDPC kodova pokrenuli su *Sipser* i *Spielman* [33] otkrićem *ekspander kodova*, kao potklase LDPC kodova sa izuzetnim teorijskim osobinama. U svom značajnom članku oni su uspeli da pokažu da se korektivne sposobnosti iterativnih dekodera, primenjenih na ekspander kodove, linearno povećavaju sa dužinom koda. Njihove rezultate generalizovali su *Barg* i *Zemor* [34, 35], dokazujući da i ekspander kodovi postižu *Shannon*-ov kapacitet, kao i *Burshtein* i *Miler* [36] koji su posmatrali iregularne LDPC kodove.

Vezu između TK memorijske arhitekture i iterativnog dekodera LDPC kodova, nazvanog *Gallager B* dekoder [26], prvi su uočili Vasić i *Chilappagari* u radu [12], gde su postavili okvir za analizu i dizajn dekodera LDPC kodova konstruisanih od nepouzdanih komponenti. Od tada su različiti autori koristeći *density evolution* analizu ispitivali asimptotske karakteristike praktično značajnih iterativnih dekodera [25, 37–42]. Sprovedene analize pokazale su da *Shannon*-ov kapacitet nije moguće dostići kada su dekoderi napravljeni od nepouzdanog hardvera. Međutim, to nije prepreka njihove primene u kodovanim memorijskim arhitekturama. Kompleksnost iterativnih dekodera raste linearno sa povećanjem dužine koda, osobina koji je *Taylor* primetio jedino kod ove vrste kodova, što je uslov koji garantuje pouzdanost memorije uz asimptotski veoma nisku kompleksnost.

1.3 Doprinos i organizacija rada

Većina referentnog istraživanja iz oblasti nepouzdanih dekodera LDPC kodova bila je usmerena ka asimptotskoj analizi, koja, kako je to primetio Vasić [43], ne opisuje adekvatno ponašanje dekodera praktično upotrebljivih kodova. Dodatno, u literaturi preovladavaju rezultati koji nepouzdanosost logičkih kola, od kojih su sastavljeni dekoderi, tretiraju kao nekorlisane događaje, prema teorijskom modelu koji je originalno predložio *von Neumann*. *Von Neumann*-ov model je samo gruba aproksimacija fizičkih procesa koji dovode do otkaza logičkih kola, i, na primer, ne opisuje adekvatno propagacione greške (eng. *timing errors*) koje dominantno utiču na nepouzdanost brzih, energetski efikasnih čipova. Smatra se da je modelovanje nepouzdanosti logičkih kola otvoren problem. U ovom radu razmatrani su modeli koji se baziraju na *Markov*-ljevim lancima, koji su se do sada u literaturi smatrali i suviše kompleksnim, pa se nisu značajnije analizirali. Međutim, samo uzimanje u obzir korelisane prirode otkaza logičkih kola dovodi do praktično upotrebljivih rezultata.

Ovaj rad izlazi iz okvira čisto asimptotskog razmatranja nepouzdanosti dekodera LDPC

kodova i pretenduje da pruži praktičnije rezultate, od do sada publikovanih u literaturi. Umesto razmatranja nekorelisanih otkaza, u ovom radu predložen je novi pristup modelovanja, koji preciznije opisuje greške koje dominanto utiču na performanse energetski efikasnih poluprovoničkih CMOS tehnologija. Uvođenje novog modela omogućilo je potpuno drugačiji pogled na nepouzdanost dekodera, pritom otvarajući mnoštvo pitanja čiji odgovori značajno menjanju dosadašnje razumevanje uticaja hardverske nepouzdanosti na performanse dekodera. Tako je u ovom radu primećeno da LDPC kodovi, iako dekodovani nepouzdanim dekoderom, imaju sposobnost garantovanog ispravljanja grešaka nastalih u toku prenosa kroz telekomunikacioni kanal, što je osobina koja je do sada bila poznata samo za pouzdane dekodere. Dugo važeće mišljenje da hardverske greške imaju samo negativan uticaj na performanse sistema, testirano je u ovom radu. Nasuprot očekivanog odgovora, koji se u literaturi navodi još od *von Neumann*-ovih pionirskih radova, ovde je pokazano da hardverska nepouzdanost može imati pozitivan uticaj na performanse iterativnih dekodera LDPC kodova. Ova iznenađujuća pojava iskošćena je za dizajniranje novih dekodera niske kompleksnosti i izuzetnih korektivnih sposobnosti. Dodatno, u ovom radu razmatrana je i primena LDPC kodova u memorijskim arhitekturama, pri čemu je predloženo rešenje koje toleriše značajan broj otkaza memorijskih komponenata.

U Poglavlju 2 ovog rada prikazan je pregled izvora kao i tipova otkaza elektronskih uređaja, sa posebnim osvrtom na otkaze koji nastaju usled smanjenja napona napajanja uređaja. Ponašanje dekodera pri ovim greškama dominantno je istraživano u ovom radu. Takođe, predloženi model otkaza verifikovan je na primeru logičkih kola praktično značajnih za implementaciju dekodera. Dodatno, predloženi su i algoritmi za probabilističku analizu nepouzdanih kombinacionih logičkih šema.

Pregled osobina i klasa LDPC kodova dat je u Poglavlju 3. Posebno su predstavljeni najznačajniji načini konstrukcija LDPC kodova korišćenih u radu, kao i univerzalni metod kodovanja. Zatim su opisani dominantno korišćeni algoritmi dekodovanja LDPC kodova, koji su poređeni prema performansama i nivoima kompleksnosti. Kao teorijski značajna klasa LDPC kodova, ekspander kodovi su izdvojeno posmatrani. Posebno je naglašena i popularna asimptotska *density evolution* analitička tehnika.

Poglavlje 4 posvećeno je analizi jednostavnih *bit-flipping* dekodera, sa posebnom pažnjom na jednokoračne dekodere bazirane na većinskom odlučivanju, čije performanse se mogu odrediti analitički. Posebno je istraživana sposobnost *bit-flipping* dekodera da garantovano ispravljaju greške, pa je pružen matematički rigorozan dokaz o postojanju kodova čija se ko-

rektivna sposobnost povećava linearno sa dužinom koda. Dodatno, broj grešaka koji se mogu ispraviti numerički je određen za različite parametre LDPC kodova.

Iznenađujuća osobina hardverskih otkaza da poboljšavaju performanse LDPC kodova, ilustrovana je u Poglavlju 5, na primeru popularnog *Gallager B* dekodera. Istražene su strukturalne karakteristike kodova koje pogoduju ovakvom ponašanju dekodera. Takođe, uspostavljen je okvir za analizu iterativnih dekodera pri korelisanim otkazima i detaljno identifikovan uticaj zavisnosti kodnih reči, koje se prenose, na performanse dekodera.

Dekoderi niske kompleksnosti, delimično konstruisani od nepouzdanih komponenti, dalje su istraživani u Poglavlju 6. Tolerantnost postojećih iterativnih dekodera LDPC kodova iskorišćena je pri konstrukciji novih kompozitnih dekodera, dobrog kompromisa kompleksnosti i korektivnih sposobnosti. Predložen je princip dekodovanja LDPC kodova, u kome se odluke donose na osnovu novog grafovskog opisa kodova, pri čemu su korektivne sposobnosti dekodera posebno istražene.

Primena LDPC kodova na povećanje pouzdanosti memorijskih elemenata istražena je u Poglavlju 7, gde je predložena memorijska arhitektura niske kompleksnosti, koja toleriše veći broj otkaza komponenti. Performanse ove memorije značajno prevazilaze rešenja na bazi *Hamming*-ovih kodova popularna u praktičnim implementacijama, čak i kada su logička kola podložna korelisanim otkazima. Ilustrovan je potencijalni pravac primene prezentovanih rezultata na aktuelne 3D polihedralne memorijske organizacije.

Zaključne napomene i potencijalni pravci daljih istraživanja dati su u Poglavlju 8.

Poglavlje 2

Nepouzdanost logičkih kola

Integrisana kola, otkrivena kasnih šezdesetih godina prošlog veka izmenila su način na koji funkcioniše moderno društvo. Nakon više od pedeset godina konstantnog napretka poluprovodničkih CMOS tehnologija integrisani čipovi ne samo da su promenili tehničke nauke, već su značajno doprineli razvoju različitih humanističkih nauka, kao što su medicina, farmakologija ili genetika. Ono što jeste bilo konstanta tokom prethodnih godina je važenje *Moore*-ovog zakona koji predviđa da se broj tranzistora koje je moguće smestiti u integrisano kolo nepromenjene površine duplira svake dve godine. Ključni parametar korišćen za opis poluprovodničkih tehnologija predstavlja dužina $gate$-a CMOS tranzistora. Tako, na primer, ako je dužina *gate*-a tranzitora $0,25$-μm, tada se kaže da je reč o $0,25$-μm-skoj tehnologiji proizvodnje tranzistora. Primetan je trend ne samo smanjenja veličine tranzistora (veća integracija poluprovodničkih komponenti), već i napona napajanja, što je ilustrovano u tabeli datoj u nastavku. Međutim,

Tabela 2.1: Trend razvoja CMOS tehnologija [2].

Godina	2006	2008	2010	2012	2014
Dužina $gate$-a (nm)	65	45	32	22	16
Napon napajanja (V)	1,2	1,0	0,9	0,8	0,7

trend gustog pakovanja tranzistora i želja za energetski efikasnim tehnologijama prirodno dovodi do njihove veće nepouzdanosti. Na primer, poznato je da je za CMOS tehnologije ispod 65-μm teško kontrolisati koncentraciju dopanta, što dovodi do poznatog problema fluktuacije atoma dopanta (eng. *Random Dopant Fluctuation*, RDF) [44]. Dodatno, prilikom utiskivanja met-

alnog sloja postoje varijacije koje dovode do hrapavosti ivica linija na čipu (eng. *Line-Edge Roughness*, LER) [45]. Takođe skaliranje napona i frekvencije negativno utiče na pouzdanost elektronskih komponenti.

U Odeljku 2.1 napravljen je pregled najznačajnijih tipova otkaza elektronskih uređaja, kao i njihovih uzroka. Otkazi logičkih kola koji su posledica smanjenja napajanja čipa u ovom radu su posebno razmatrani, pa je ovom tipu otkaza posvećen Odeljak 2.2. Pojava otkaza dominantno zavisi od promena vrednosti na izlazu iz logičkog kola, koje zahtevaju disipaciju najveće količine energije. Predloženi model stanja (eng. *state model*) ovih otkaza verifikovan je na primeru logičkih kola praktično značajnih za implementaciju dekodera kodova male gustine provera parnosti. U Odeljku 2.3 prezentovane su osnove probabilističke analize logičkih koja, koja se može koristiti za procenu pouzdanosti nekog elektronskog uređaja. Predložena su dva algoritma koja uzimaju u obzir vremensku korelaciju otkaza. Neke zaključne napomene date su Odeljku 2.4.

2.1 Problem nepouzdanosti elektronskih sistema

Nepouzdanost poluprovodničkih tehnologija manifestuje se kroz otkaze pojedinih elektronskih komponentni napravljenih u datoj VLSI (eng. *Very Large-Scale Integration*) tehnologiji. Prema trajanju otkaze je moguće podeliti na i) trajne (eng. *permanent*), ii) naizmenične (eng. *intermittent*) i iii) tranzijentne (eng. *transient*). Trajni (permanentni ili tvrdi) otkazi nastaju kao posledica nesavršenosti proizvodnog procesa čipa i manifestuju se kroz nemogućnost tranzistora da trajno promeni izlaznu vrednost. Tada se obično kaže da se komponenta "zaglavi" (eng. *stuck-at*) i trajno prestaje da funkcioniše. Naizmenične greške obično prethode trajnim i deo su prelažnog režima koji nastupa pre neko što komponenta trajno otkaže. Tranzijentni (meki) otkazi se javljaju povremeno, slučajno, pri čemu se njihova pojava može opisati probabilistički. Sa stanovišta teorije informacija ovi otkazi su najinteresantniji, jer omogućavaju prenos informacija iako je izvor informacija korumpiran. Može se smatrati da postoje tri značajna izvora mekih grešaka i to i) visoko-energetske alfa čestice kosmičkog zračenja, ii) varijacije u procesu proizvodnje i iii) smanjenje napona napajanja.

Kosmičko zračenje se povećava sa udaljavanjem od nivoa zemlje i dugo se smatralo da udari alfa čestica utiču na energetski nivo tranzistora samo na velikim visinama. Posebno su bili ugroženi elektronski uređaji na avionima, pa su primećeni i slučajevi padova aviona

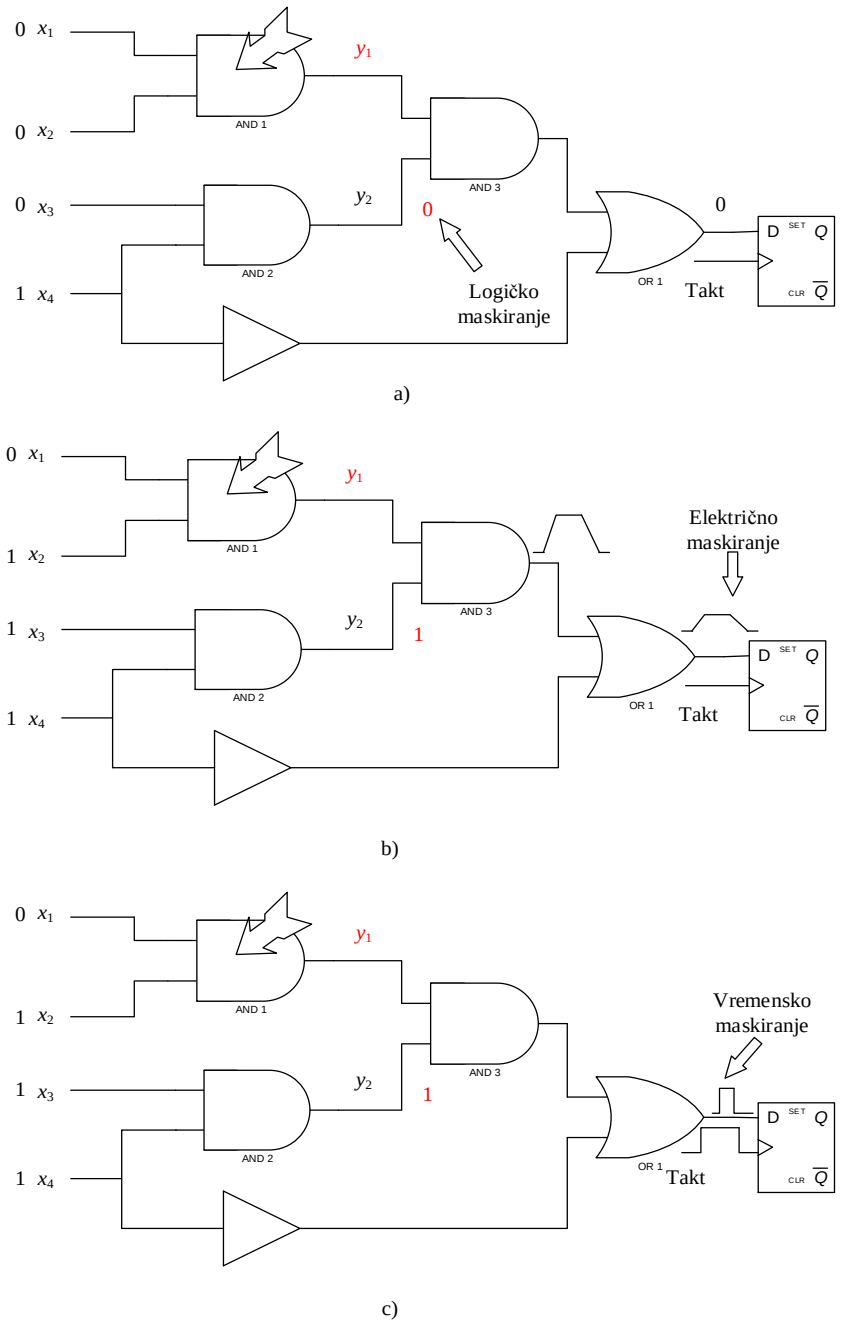
uzrokovani elektronskim otkazima. Međutim, sa smanjnjem tehnologije izrade i kao i napona napajanja udari alfa čestica izazivaju otkaze elektronskih komponenti smeštenih na zemlji.

Varijacije u procesu proizvodnje čipova dovode do neuniformne raspodele napona praga uključivanja tranzistora. Tako viši napon praga uključivanja nekih tranzistora od nominalnog, pri fiksnom naponu napajanja, može dovesti do sporijeg uključivanja tranzistora od očekivanog, što će se manifestovati otkazom logičkog kola.

Konačno, smanjenje napona napajanja dovodi do duže stabilizacije signala na izlazu iz nekog logičkog kola. Ako se to kolo nalazi na kritičnoj putanji može doći do otkaza. Bilo da se napon smanjuje namerno (skaliranje napona) ili dolazi do slučajnih padova napona, margina za šum se smanjuje. Ovaj tip grešaka biće detaljno opisan u narednom odeljku.

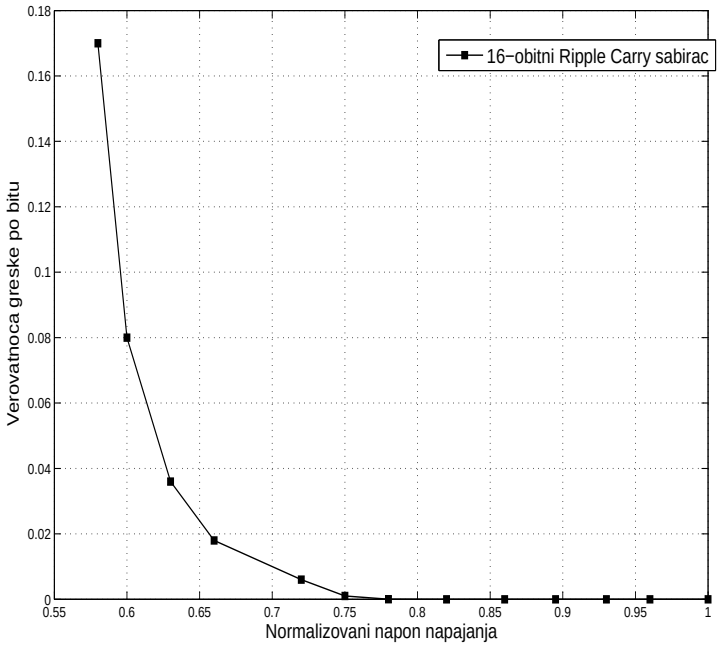
Dugo se više značaja pridavalo otkazima memorijskih elemenata nego otkazima logičkih kola. Razlozi za to vezani za takozvano maskiranje otkaza koje može biti logičko, električno ili vremensko (temporalno). Na slici 2.1.a ilustrovano je logičko maskiranje otkaza kola AND1. Neka je, recimo zbog udara alfa čestice, AND1 kolo otkazalo i neka se na njegovom izlazu javi pogrešna vrednost 1. Konfiguracija kola je takva da ova greška neće dalje propagirati, tj. biće maskirana vrednossću 0 koja se nalazi na izlazu iz kola AND2. Ako je, s druge strane, konfiguracija ulaznih vrednosti takva da otkaz kola neće biti logički maskiran, postoji nenulta verovatnoća da se u flip-flop registar upiše pogrešna vrednost. Međutim, može se desiti da se vrednost napona signala greške postepeno smanjuje kako signal dalje propagira kroz graf kola, što će smanjiti verovatnoću upisa pogrešne vrednosti u krajnjem registru. Opisani efekat naziva se električnim maskiranjem i ilustruvan je na slici 2.1.b. Dodatno, može se desiti i da se greška ne registruje zato što ne zadovoljava uslove vremenske stabilizacije signala, tj. pogrešna vrednost će se javiti tek nakon upisa signala u registar. Tada govorimo o vremenskom maskiranju otkaza prikazanom na slici 2.1.c. Međutim, treba primetiti da je na sličan način moguće i maskiranje ispravnih vrednosti signala. Kako su logička kola sve nepouzdanija, maksiranje otkaza ne nudi dovoljnu zaštitu i problemima nepouzdanosti se pridaje sve veći značaj. Tako je na slici 2.2 ilustovan drastičan uticaj smanjenja napona na pouzdanost 16-bitnog *Ripple Carry* sabirača implementiranog u 135-nm IBM tehnologiji. Rezultat je preuzet iz veoma informativnog rada [1], u kome se čitalac takođe može informisati o uticaju procesnih varijacija, kao i propagacionog kašnjenja na pouzdanost kombinacionih logičkih mreža.

Prvi korak u analizi nepouzdanosti logičkih kola je modelovanje otkaza. Raznovrsnost izvora otkaza otežava pronalazak unificiranog matematičkog modela, koji bi objedinio veći broj



Slika 2.1: Ilustracija a) logičkog, b) električnog i c) vremenskog maskiranja otkaza logičkih kola.

izvora otkaza. Neki otkazi su zavisni od informacione (bitske) sekvence koja se obrađuje, dok su drugi vremenski nekorelisani, ali pogađaju susedna logička kola (prostorno su korelisani). Analiza velikih logičkih kola na tranzistorskom nivou (koji bi najbolje ilustrovao problem nepouzdanosti) je i suviše kompleksna, pa je uobičajno da se problem nepouzdanosti rešava na višem nivou apstrakcije – logičkom nivou, ili čak sistemskom (eng. *Register Transfer Layer*, RTL) nivou gde se otkazi dodaju samo na izlazima većih kombinacionih blokova. Specijalno, u



Slika 2.2: Nepouzdanost 16-obitnog sabirača kao posledica smanjenja napajanja (nominalni napon napajanja 1,35V) [1].

dostupnoj literaturi hardverski otkazi u kontekstu dekodera kodova sa malom gustinom provera parnosti najčešće su modelovani takozvanim *von Neumann*-ovim modelom otkaza, nazvanim u čast matematičara koji je prvi istraživao problem pouzdanosti sistema napravljenih od nepouzdanih komponenti [13]. Ovaj model podrazumeva da su otkazi logičkih kola vremenski i prostorno nekorelisani, tj. da svako kolo otkazuje sa unapred poznatom fiksnom verovatnoćom, opisanom *Bernoulli*-jevom raspodelom. Model je isuviše pojednostavljen da bi adekvatno opisivao i jedan značajan izvor otkaza, ali svoju popularnost duguje pre sve jednostavnosti. Jedan od ciljeva ovog rada je i predlog nove paradigme modelovanja otkaza, koji bi pružio rezultate sa više praktičnog značaja, od do sada dostupnih (eng. *state-of-the-art*) istraživanja.

Usled gustog pakovanja tranzistora u memorijskim elementima, RDF i LER efekti dovode do curenja naboja napajanja memorijskih ćelija, kašnjenja u pristupu memoriji, kao i razmicanja (eng. *mismatch*) praga uključivanja tranzistora. Tako, na primer, u SRAM (eng. *Static Random-Access Memory* otkazi memorijskih ćelija mogu nastati kao posledica [46]

1) destruktivne operacije čitanja, kada se bit koji treba pročitati invertuje (eng. *read failure*);

2) neuspešnog upisa na memorijsku lokaciju (eng. *write failure*);

3) povećanja vremena pristupa memoriji koje dovodi do narušavanja pristupa memoriji (eng. *access-time failure*);

4) narušavanje sadražaja memorijske ćelije usled smanjenja napajanja memorije u trenutku kada se ne pristupa memoriji (eng. *hold failure*).

Autori referentnog rada [46] pojedinačno su razmatrali sva četiri tipa memorijskih otkaza, pri čemu su razmicanja pragova uključivanja tranzistora modelovali kao nezavisne *Gauss*-ove promenljive, što su opravdali činjenicom da RDF varijacije zavise samo od geometrije tranzistora, a ne i od položaja tranzistora na čipu [47]. Prostornu korelaciju potrebno je uključiti u analizu ako bi se posmatrali efekti varijacija dužine ili debljine kanala tranzistora. U ovom radu je takođe razmatran uticaj RDF na pouzdanost memorijskih ćelija, pa se prostorna korelacija otkaza memorijskih ćelija zanemarivala. Podrazumevani su tranzijentni otkazi, pri čemu se nepouzdanost manifestuje invertovanjem vrednosti skladištene u ćeliji sa nekom verovatnoćom, u trenutku čitanja vrednosti iz memorije. Drugim rečima, otkaz ćelije ne degradira ćelijsku strukturu, već samo utiče na sadržaj ćelije.

Problem nepouzdanosti memorijskih uređaja je u literaturi poznat i istraživan u prošlosti, tako da se pojavile i monografije posvećene toj temi. Jedna od njih [48] ispituje pogodnost upotrebe linearnih blok kodova u NAND i NOR *flesh* memorijama. U operacionim memorijama *Hamming*-ovi kodovi su odavno prestavljeni kao standardno rešenje. U ovom radu predloženo je rešenje bazirano na kodovima sa malom gustinom provera parnosti, koje obezbeđuje pouzdanost memorije čak i kada su logička kola koja služe za ažuriranja memorijskih lokacija nepouzdana. Za više informacija čitalac se upućuje na Poglavlje 7.

2.2 Otkazi kao posledica smanjena napajanja logičkih kola

Smanjenje napona napajanja u cilju konstrukcije energetski efikasnih CMOS čipova dovodi do pojave grešaka u logičkim kolima implementiranim na čipu [49]. Dodatno, različite tehnike skaliranja napona i frekvencije, kao na primer AVS (eng. *aggressive voltage scaling*) [50], smanjuju otpornost poluprovodničkih struktura na greške. Greške u memorijskim ćelijama koje nastaju zbog smanjenja napajanja su usled RDF nezavisne i prostorno nekorelisane [51]. S druge strane, propagaciona kašnjenja između aritmetičkih jedinica ili logičkih kola su zavisna od ulaznih vrednosti u logičke blokove i teško je pronaći izraze u zatvorenoj formi pomoću kojih je moguće proceniti pouzdanost kompozitnih (većih) logičkih blokova. Smanjenje napona

napajanja dovodi do sporije stabilizacije napona na izlazu iz logičkog kola, pa se verovatnoća da se vrednost signala odabira (koristi u narednom logičkom bloku) pre nego što se stabilizuje povećava [3]. Da bi se adekvatno opisali otkazi logičkih kola u ovom radu predložen je generalni *model stanja* koji uzima u obzir vremensku zavisnost otkaza.

Neka je $f : \{0,1\}^m \rightarrow \{0,1\}$, $m > 1$, *Boole*-ova funkcija sa m argumenata, koja u trenutku k generiše izlaz $z^{(k)} = f(y_1^{(k)}, y_2^{(k)}, \ldots, y_m^{(k)})$, gde je sa $\mathbf{y}^{(k)} = [y_1^{(k)}, y_2^{(k)}, \ldots, y_m^{(k)}]$ označen vektor ulaznih argumenata u trenutku k. Kako funkciju obavlja nepouzdano logičko kolo to se stvarni izlaz logičkog kola u trenutku k može predstaviti kao

$$\hat{z}^{(k)} = f(y_1^{(k)}, y_2^{(k)}, \ldots, y_m^{(k)}) \oplus \xi^{(k)}, \tag{2.1}$$

gde vrednost greške u trenutku k, $\xi^{(k)} \in \{0,1\}$, zavisi od M uzastopnih vektora argumenata $y_1^{(k-M+1)}, \ldots, y_m^{(k-M+1)}, \ldots, y_1^{(k)}, \ldots, y_m^{(k)}$, koji formiraju *stanje* u trenutku k, označeno sa $\mathbf{s}^{(k)} = \{\mathbf{y}^{(j)}\}_{j \in [k-(M-1),k]}$. Predloženi model se naziva i "mutantskim" jer izlaz kola mutira sa nekom verovatnoćom. Vrednost $\xi^{(k)}$ je partikularna realizacija slučajne promenljive Ξ, koja se može izraziti

$$\Pr\{\xi^{(k)} = 1\} = \int_X^{+\infty} w_\Xi(x; \mathbf{s}^{(k)}) dx, \tag{2.2}$$

gde $w_\Xi(x, \mathbf{s}^{(k)})$ raspodela slučajne promenljive Ξ u odeđenom stanju, x označava tehnološki parametar čije varijacije dovode do otkaza logičkog kola, X je prag korišćen u konkretnoj implementaciji, dok se efekti koje izazivaju različiti ulazi u logičko kolo uzimaju u obzir preko parametara funkcije raspodele, kao na primer, varijanse, ili parametra oblikovanja.

Izbor gustine raspodele zavisi od parametra x koji izaziva otkaz kola, i može se odnositi na napon napajanja, propagaciono kašnjenje signala, ili prag uključivanja tranzistora. U opštem slučaju, kada se izabere parametar x, gustinu raspodele moguće je proceniti merenjima ili simulacijama izabrane poluprovodničke tehnologije. Na primer, u [3] je predložen matematički model procene propagacionog kašnjenja vremenski sinhronih kola. Autori navedenog članka su uočili da se vreme potrebno za stabilizaciju signala na izlazu iz kola, koja se napajaju naponom manjim od nominalnog, može adekvatno opisati inverznom *Gauss*-ovom gustinom raspodele date sa

$$w_\Xi(x; \mu^{(k)}, \lambda^{(k)}) = \sqrt{\frac{\lambda^{(k)}}{2\pi x^3}} e^{\frac{-\lambda^{(k)}(x-\mu^{(k)})^2}{2(\mu^{(k)})^2 x}}, x \geq 0, \tag{2.3}$$

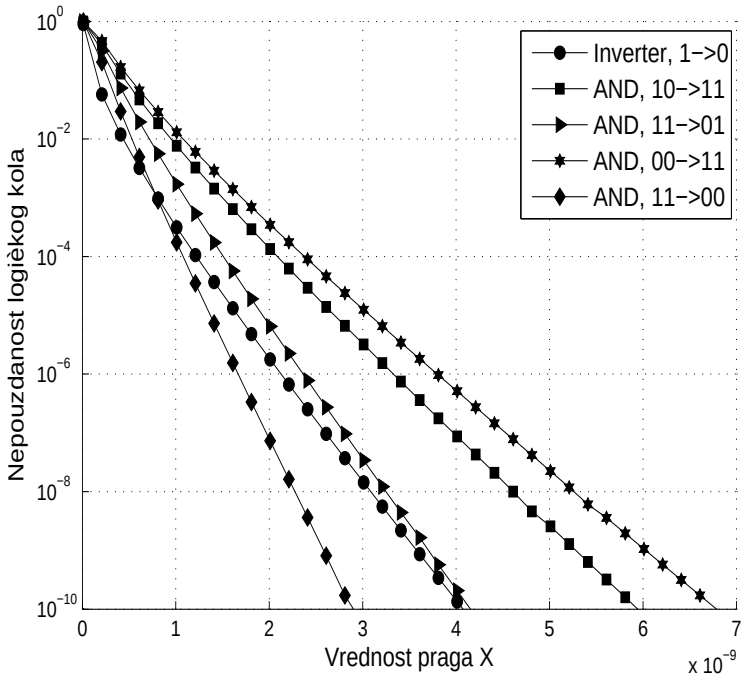
gde $\mu^{(k)}$ predstavlja srednju vrednost raspodele, dok je $\lambda^{(k)}$ parametar oblikovanja funkcije raspodele pridružen trenutku (stanju) k. Uočava se da greške povezane sa propagacijom signala zavise samo od dva sukcesivna trenutka posmatranja, prevedeno u naš model važi $M = 2$.

Vrednosti $\mu^{(k)}$ i $\lambda^{(k)}$ zavise od stanja $\mathrm{s}^{(k)}$i mogu se proceniti empirijski. Dodatno, oni tipično zavise od posmatranog logičkog kola. U radu [3] posmatrana su složena logička kola koja se sastoje od invertera i dvoulaznih AND logičkih kola, što odgovara poznatoj AIG (eng. *AND-Inverter Graph*) predstavi. U tabeli datoj u nastavku prikazane su vrednosti parametara funkcije gustine raspodele za nekoliko ulaznih tranzicija. Prikazane vrednosti imaju za cilj samo da ilustruju trendove nepouzdanosti, pa su tehnički detalji korišćeni u analizi na ovom mestu izostavljeni.

Tabela 2.2: Parametri funkcije gustine raspodele invertora i AND logičkog kola [3].

Logičko kolo	$\mathbf{s}^{(k)}$	$\mu^{(k)}$	$\lambda^{(k)}$
Invertor	$1 \rightarrow 0$	0.65×10^{-10}	0.36×10^{-10}
AND	$10 \rightarrow 11$	2.3×10^{-10}	3.4×10^{-10}
AND	$11 \rightarrow 01$	1.9×10^{-10}	3.4×10^{-10}
AND	$00 \rightarrow 11$	2.6×10^{-10}	3.8×10^{-10}
AND	$11 \rightarrow 00$	1.5×10^{-10}	3.1×10^{-10}

Na osnovu gustine raspodele propagacionog kašnjenja, verovatnoće otkaza logičkih kola moguće je lako proceniti. Na slici 2.3 numerički su prikazane verovatnoće otkaza u funkciji praga X, koji označava vremenski period dodeljen kolu za proces odlučivanja. Vrednost praga je fiksna za specifičnu hardversku realizaciju. Ako je vreme uspostavljanja stabilne vrednosti signala veće od X izlaz logičkog postaje pogrešan. Očigledno je da prolongiranje donošenja odluke povećava njenu pouzdanost. Posmatranjem slike zaključuje se takođe da iako napravljna u istoj tehnologiji različita logička kola ostvarauju značajno različite nivoe pouzdanosti. Na primer, ako je $X = 3ns$, verovatnoća otkaza invertera iznosi približno 10^{-8}, dok nepouzdanost AND logičkih kola može iznositi 10^{-5}. Dodatno, primećuje se da verovatnoće otkaza zavise od stanja u kome se kolo nalazi, pa nepouzdanost može varirati i za red veličine. U prezentovanom primeru tranzicija ulaznih vrednosti $11 \rightarrow 00$ dovodi do najmanje nepouzdanosti AND logičkog kola. Primetiti da je otkaz logičkog kola moguć samo kada logičko kolo menja izlaz, što je u saglasnosti sa poznatim stavom da energetska potrošnja logičkog kola linearno raste sa povećanjem broja promena izlazne vrednosti signala (eng. *switching activity*) [52]. Tada se raspodela ulaznih tranzicija može zameniti raspodelom izlaznih tranzicija.



Slika 2.3: Verovatnoće otkaza logičkih kola.

Neka je sa $T_y^{i\to j}$ raspodela kašenjenja signala na izlazu y kada se izlaz kola menja sa i na j, $i, j \in \{0, 1\}$. Ako se izlaz ne menja tada je propagaciono kašnjenje jednako nuli, što se može predstaviti delta funkcijom $\delta(t)$ kao

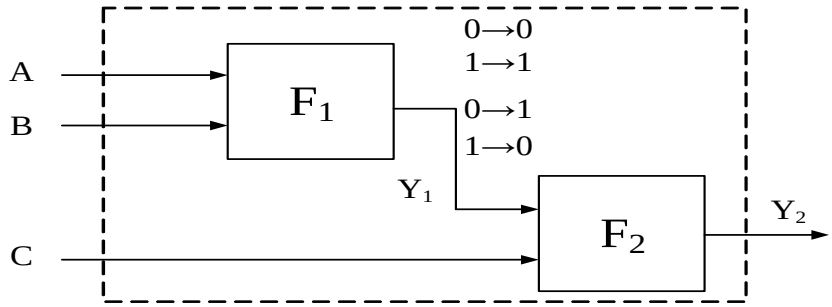
$$T_y^{0\to 0} = \delta(t), \;\; T_y^{1\to 1} = \delta(t). \tag{2.4}$$

S druge strane, gustine raspodela za dve preostale izlazne tranzicije ($0 \to 1$ i $1 \to 0$) mogu se odrediti na sledeći način

$$T_y^{0\to 1} = \sum_k \beta_k^{0\to 1} \times IG(\lambda^{(k)}, \mu^{(k)}), \tag{2.5}$$

$$T_y^{1\to 0} = \sum_k \beta_k^{1\to 0} \times IG(\lambda^{(k)}, \mu^{(k)}), \tag{2.6}$$

gde je sa $IG(\lambda^{(k)}, \mu^{(k)})$ označena inverzna *Gauss*-ova raspodela data u jednačini (2.3), a $\beta_k^{i\to j}$ predstavlja verovatnoću pojave k-te kombinacije ulaza koja dovodi do izlazne tranzicije $i \to j$. Kako se kompozitna logička kola mogu predstaviti kao grafovi (kaskade) čiji su čvorovi dvoulazna (ili jednoulazna) kola, raspodele $T_y^{0\to 1}$ i $T_y^{1\to 0}$ potrebno je propagirati od primarnih ulaza dalje kroz graf kola. Na slici 2.4 prikazano je 3-ulazno logičko kolo čiji su primarni izlazi označeni sa A, B i C, dok je Y_2 primarni izlaz kola. Profil propagacionog kašnjenja



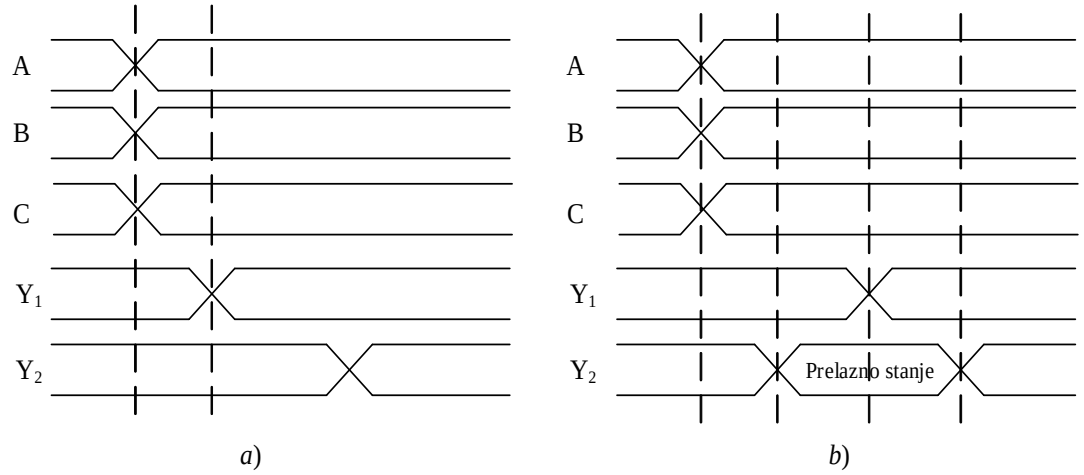
Slika 2.4: Kaskadna šema logičkog kola.

nekog od internih signala na izlazu iz logičkog kola dobija se konvolucijom raspodela kašnjenja dva ulazna signala. Postupak se nastavlja dok se ne dostignu primarni izlazi, kada se formira konačna raspodela kašnjenja izlaznih signala. Navedeni postupak ne uzima u obzir dva značajna fenomena i to i) *prelazna stanja uzrokovana tranzicijama internih signala* i ii) *korelaciju internih signala*. Prvi fenomen odnosi se na moguće formiranje prelaznih stanja kao posledice različitih propagacionih kašnjenja po internim putanjama signala. Na slici 2.5 ilustrovana su dva slučaja i to jedan u kome ne dolazi do prelaznog stanja i drugi u kome prelazno stanje postoji. Neka su na primer logička kola označena sa F_1 i F_2 2-ulazna XOR logička kola i neka u k-tom trenutku primarni ulazi u logičko kolo uzimaju vrednosti $(A, B, C) = (1, 0, 0)$, što dovodi do izlazne vrednosti $Y_2 = 1$. Neka u narednom trenutku važi $(A, B, C) = (1, 1, 1)$, što potencijalno ne menja primarni izlaz kola. Međutim, ako se vrednost signala C promeni brže od signala B ili se zbog pada napona računanje signala Y_1 produži, u kraćem vremenskom intervalu (prelaznom stanju) važiće $Y_2 = 0$. Nakon što i signal B promeni vrednost primarni signal Y_2 će se stabilizovati na ispravnoj vrednosti. Međutim, ako se odluka o signalu donese u toku trajanja prelaznog stanja, ona će biti pogrešna. Prelazna stanja se mogu javiti samo kada više od jednog ulaza promeni vrednost.

U radu [53] pokazano je kako se prelazna stanja mogu uključiti prilikom formiranja profila kašnjenja. Ono što treba uraditi je uračunati i relativna kašnjenja između dva logička kola (F_1 i F_2 u našem primeru). Tako je, na primer, primećeno u [53] da je raspodelu kašenja izlazne tranzicije $1 \rightarrow 0$ signala Y_2 moguće proceniti kao

$$T_{Y_2}^{1\rightarrow 0} = \sum_{i,j} T_{Y_1}^{1\rightarrow 0} \star (1 + h(\mu^{(i)})) \times IG(\lambda^{(j)}, \mu^{(j)}), \tag{2.7}$$

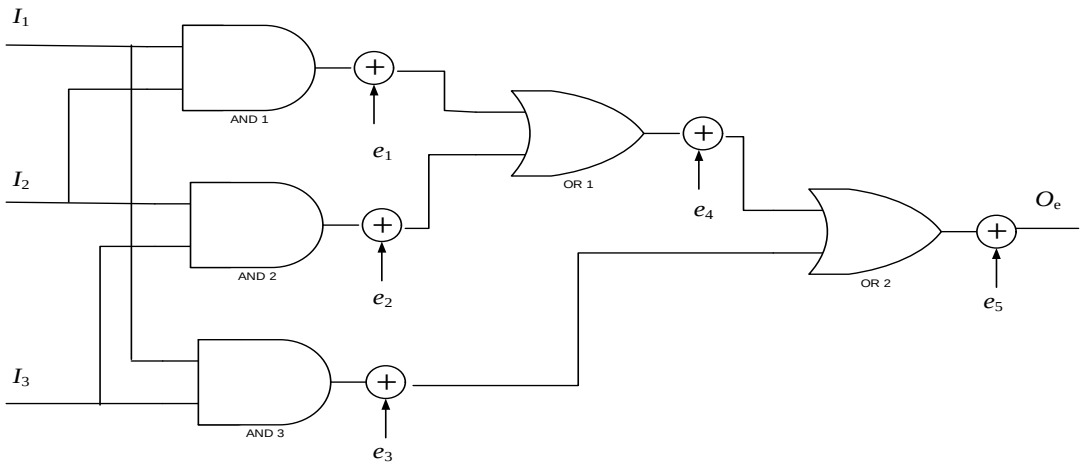
gde se $T_{Y_1}^{1\rightarrow 0}$ izračunava preko formule (2.6), $h(\mu^{(i)})$ odgovara *Heaviside*-ovoj odskočnoj funkciji sa odskokom u tački srednje vrednosti gustine raspodele kola F_2, $\mu^{(i)}$, dok je sa $\star$ označen operator konvolucije.



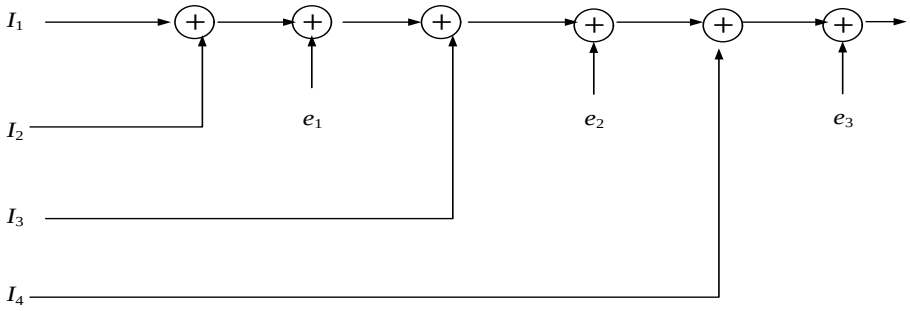
Slika 2.5: Vremenski dijagram (eng. *timing*) logičkog kola sa slike 2.4: a) bez prelaznih stanja b) kada postoje prelazna stanja.

Propagiranje raspodela kašenjena ne uzima u obzir međusobnu zavisnost internih signala, i predloženom metodom moguće je proceniti samo logička kola čiji graf predstavlja stablo. Ako u grafu kaskadnog kola postoje račvanja (eng. *reconvergent fan-out*), tako da se vrednosti jednog signala propagiraju po više putanja i kasnije sažimaju u jedinstvenom čvoru grafa, predloženi metod neće adekvatno proceniti raspodelu kašnjenja.

Drugačiji pristup podrazumeva da se odluka o otkazu komponentnog logičkog kola donosi lokalno na izlazima komponentnih logičkih kola, što predstavlja već uvedeni mutantski pristup prilagođen izlaznim tranzicijama. Na slici 2.6 predstavljene su arhitekture 3-ulaznog kola za većinsko odlučivanje (eng. *MAJority logic gate*, MAJ) kao i 4-ulaznog XOR kola. Sekvence grešaka e_1, e_2, e_3, e_4, e_5 su binarne, pri čemu se jedinice u sekvencama javljaju sa unapred zadatom verovatnoćom samo kada dolazi do promene vrednosti izlaza kola. Kako će to biti predstavljeno u narednom odeljku, ovakav model moguće je analitički ispititivati, kada je veličina kaskadnog kola relativno mala. Međutim, kako se iterativni dekoderi kodova sa malom gustinom provera parnosti svakako ne mogu smatrati malim logičkim kolima, ni ovakav pristup nije potpuno adekvatan. Treba naglasiti da ni Monte Karlo simulaciona analiza ne predstavlja dobro rešenje, pre svega zbog visokih procesorskih zahteva i dugog vremena potrebnog za dobijanje upotrebljivih rezultata. Da bi se izveli generalni zaključci m-ulazno kolo potrebno je simulairati za svaku od 2^m ulaznih kombinacija. Takođe, dužina simulacije nelinearno se povećava sa povećanjem pouzdanosti logičkog kola, pa je, na primer, teško proceniti nepouzdanost kompozitnog logičkog kola, koje sadrži veći broj elemenata čija je nepouzdanost $< 10^{-4}$.



a)



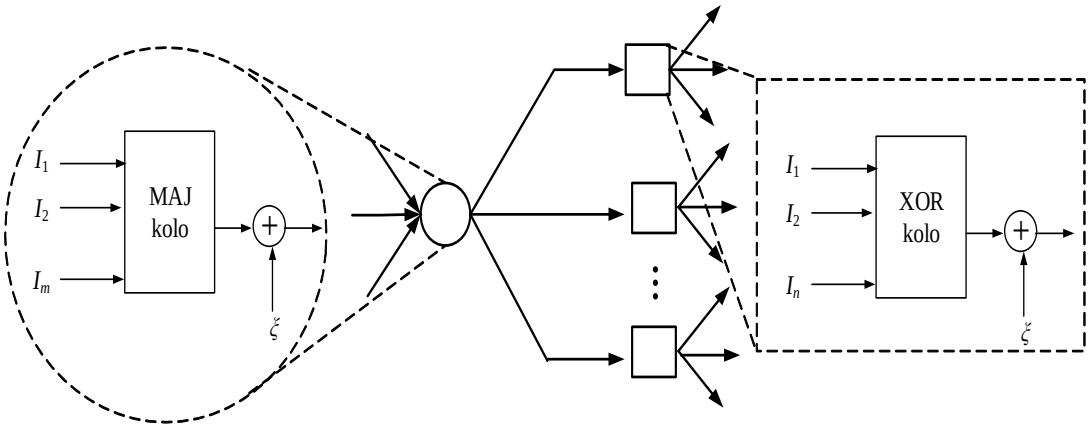
b)

Slika 2.6: Kaskadna arhitektura i mutantski model otkaza a) 3-ulaznog kola za većinsko odlučivanje (MAJ) i b) 4-ulazno XOR kola.

Zbog svega navedenog, modelovanje otkaza potrebno je dodatno pojednostaviti. Na tragu *Einstein*-ovog zapažanja da model kojim se opisuju pojave treba da bude najjednostavniji mogući (ali ne i jednostavniji od toga!), i da pruži uvid u generalnije trendove na račun preciznosti, u ovom radu predložen je GOS (eng. *Gate-Output Switching*) model otkaza logičkih kola koja nastaju kao posledica smanjenog napona napajanja. Prema GOS modelu VLSI dizajn se rastavlja na manja *jedinična* logička kola, koja umesto da se dalje rastavljaju na 2-ulazne jedinice, posmatraju kao kompaktna celina. Model je ilustrovan na slici 2.7 gde su jedinična logična kola MAJ i XOR logičkih kola, koja su sastavni deo praktično značajnih dekodera. Izlaz jediničnog kola se mutira uz pomoć sekvence grešaka ξ, pri čemu promena izlazne vrednosti može dovesti do otkaza, pa važi [54]

$$\Pr\{\xi^{(k)} = 1 | z^{(k)} \neq z^{(k-1)}\} = \varepsilon, \quad \varepsilon > 0. \tag{2.8}$$

Vrednost ε se može proceniti merenjima ili simulacijama izabrane poluprovodničke tehnologije



Slika 2.7: Pojednostavljeni GOS model otkaza.

i u ovom radu se obično smatra da je ona poznata dizajneru dekodera.

S druge strane, kada se izlaz u dva sukcesivna trenutka ne menja smatra se da je propagaciono kašnjenje jednako nuli i ne dolazi do otkaza jediničnog logičkog kola, odnosno

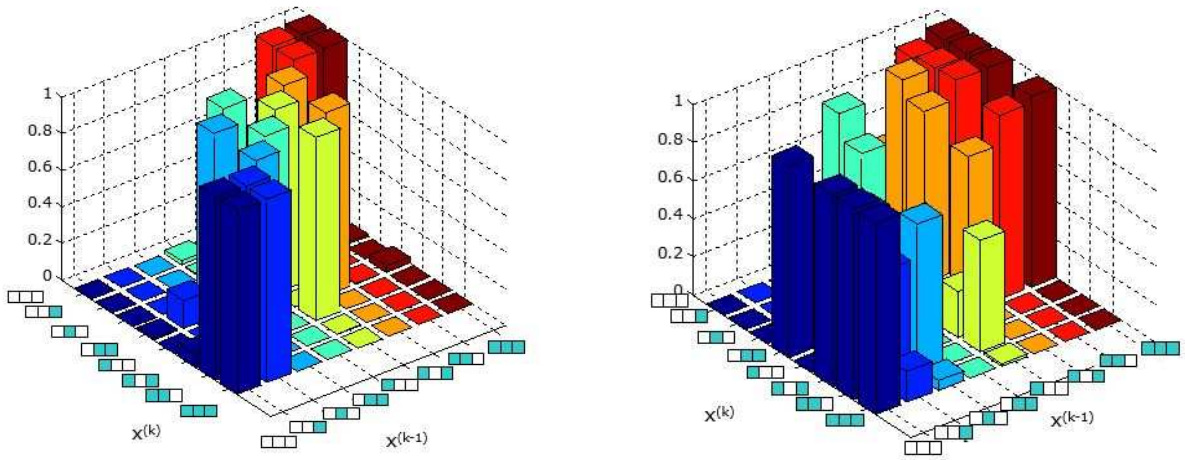
$$\Pr\{\xi^{(k)} = 1 | z^{(k)} = z^{(k-1)}\} = 0. \tag{2.9}$$

Na preciznost GOS model otkaza direktno utiče izbor jediničnih logičkih kola; kada su jedinična kola 2-ulazna model je najprecizniji. GOS model uzima u obzir korelaciju internih signala, kao i efekte logičkog maskiranja otkaza povezanog sa funkcionalnim karakteristikama kola. GOS model ne može da opiše prelazna stanja na izlazima iz jediničnih kola i neće registrovati grešku koja nastaje kao posledica tranzicije signala unutar jediničnih logičkih kola.

Iterativni dekoderi razmatrani u ovom radu sastavljeni su od velikog broja m-ulaznih XOR i MAJ logičkih kola, pa su upravo ova kola izabrana da budu jedinična. GOS model je verifikovan na primerima 3-ulaznih MAJ i XOR kola, implementiranim u 45nm-skoj CMOS tehnologiji, određivanjem pouzdanosti koju postižu za različite ulazne tranzicije (slika 2.8). Logička kola su napajana naponom koji je za 50% niži od nominalnog (0,5 VDD), dozvoljeno vreme stabilizacije signala iznosilo je 55ps, dok su varijacije napona napajanja, kao i praga uključivanja tranzistora opisane normalnom raspodelom.

U toku simulacija nije primećeno postojanje prelaznih stanja i situacijama kada se izlaz kola nije menjao logička kola su radila pouzdano. Promena izlaza je sa određenom verovatnoćom dovodila do otkaza kola, pri čemu se verovatnoća razlikovala za različite ulazne tranzi-

[1] Simulacije obavio J. Chen sa Univerziteta u Korku na osnovu metodologije prezentovane u [3]; MAJ logičko kolo je implementirano kao standardno rešenje iz *Cadence* digitalne biblioteke, dok je XOR kolo implementirano kao serijska veza dva dvoulazna XOR kola.



(a) (b)

Slika 2.8: Pouzdanost 3-ulaznih a) XOR i b) MAJ logičkih kola [1].

cije. Dodatno treba istaći da je zbog principa funkcionisanja iterativnih dekodera verovatnoća istovremene promene više ulaza XOR ili MAJ kola mala, što dodatno smanjuje uticaj prelaznih stanja. Potvrdu prethodnog zapažanja moguće je pronaći u Poglavlju 5, posvećenom *Gallager B* dekoderu. GOS model se može smatrati optimističnim, jer dozvoljava postojanja stanja u kojima ne može doći do otkaza logičkog kola. S druge strane, *von Neumann*-ov model koji podrazumeva vremensku nekorelisanost otkaza je suviše pesimističan i na osnovu svega navedenog može se zaključiti da ne opisuje adekvatno otkaze nastale kao posledica smanjenja napona napajanja. Može se smatrati da *von Neumann*-ov model omeđava nepouzdanost logičkog kola sa gornje strane, dok GOS model pruža donju granicu nepouzdanosti, pri čemu ona značajno više odgovara realnoj situaciji.

Na kraju treba istaći da neki značajni izvori otkaza ne mogu biti opisani kako GOS tako ni *von Neumann*-ovim modelima. Pre svega reč je o bombardovanju alfa česticama, koje najčešće generišu otkaze susednih logičkih kola. Njihov matematički opis bi bio značajno kompleksniji, jer mora uključitati i prostornu korelisanost otkaza. Za više informacija o uticaju alfa čestica na nepouzdanost logičkih kola čitaocu se preporučuju članci [55, 56]. Takođe, različiti otkazi nastali usled nesavršenosti proizvodnog procesa poluprovodnihkih komponenata mogu imati drugačiju statistiku.

2.3 Probabilistička analiza nepouzdanih logičkih kola

Probabilistička analiza logičkih kola ima za cilj da odredi verovatnoću da izlaz nekog kola uzima vrednost "1". odnosno "0". Obično se analiza umerava ka proceni verovatnoće da određeni signal x uzima vrednost "1", odnosno $\Pr\{x = 1\} = p(x)$.

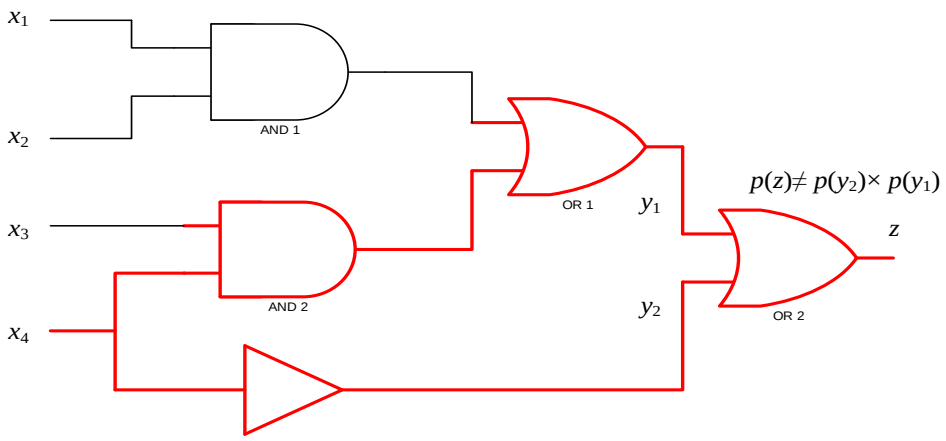
U vremenu kada analiza pouzdanosti logičkih kola nije bila u fokusu istraživanja, probabilistička analiza pružala je indikacije koliko je teško testirati i kontrolisati pojedine signale [57]. Tako se pouzdana logička kola ovom metodom mogu sa dosta uspeha analizirati, kao je to prikazanu u referentnim člancima [58–60]. Od posebno interesa je *Parker-McCluskey* metod [58] koji omogućava simboličku analizu logičkih kola na osnovu jednostavnih probabilističkih pravila datih u tabeli u nastavku. Pravila definišu verovatnoću pojave signala z na izlazu n-ulaznog elementarnog logičkog kola, $p(z)$, kao funkciju verovatnoća ulaznih signala $x_1, x_2, \ldots, x_n$, $p(x_i)$, $1 \leq i \leq n$.

Tabela 2.3: Pravila probabilističke analize elementarnih n-ulaznih logičkih kola.

Logičko kolo	Pravilo
NOT	$p(z) = p(x)$
AND	$p(z) = \prod_{i=1}^{n} p(x_i)$
NAND	$p(z) = 1 - \prod_{i=1}^{n} p(x_i)$
OR	$p(z) = 1 - \prod_{i=1}^{n} (1 - p(x_i))$
NOR	$p(z) = 1 - \prod_{i=1}^{n} p(x_i)$

Neka je dato m-ulazno kompozitno kolo, čiji su primarni ulazi označeni sa $x_1, x_2, \ldots, x_m$. Na osnovu probabilističkih pravila verovatnoće $p(x_1), p(x_2), \ldots, p(x_m)$ se propagiraju kroz graf kola dok se verovatnoća primarnog izlaza z, $p'(z)$, ne izrazi kao funkcija ovih verovatnoća, tj. $p'(z) = f(p(x_1), p(x_2), \ldots, p(x_m))$. Međutim, ovako izračunata verovatnoća ne predstavlja tačno rešenje ako postoji međusobna korelacija internih signala. Na slici 2.9 ilustrovano je postojanje korelacije koja otežava probabilističku analizu. *Parker* i *McCluskey* su primetili da je korelacija u izrazu $p'(z)$ izražena preko stepena $(p(x_i))^j$, $j > 1$, $1 \leq i \leq m$. Ako se stepeni jednostavno izbrišu dobija se

$$p(z) = p'(z)|_{(p(x_i))^j = p(x_i)}, \quad j > 0, 1 \leq i \leq m, \tag{2.10}$$



Slika 2.9: Međusobna korelacija internih signala kao problem procene verovatnoće izlaznog signala.

što predstavlja tačnu vrednost verovatnoće izlaznog signala.

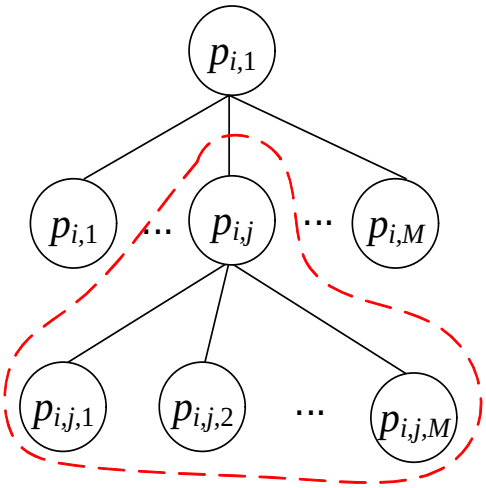
Ako se otkazi logičkih kola smatraju nekorelisanim, originalni *Parker-McCluskey* algoritam je moguće lako proširiti i na nepouzdana logička kola. Kako se nepouzdanost logičkih kola modeluje invertovanjem izlazne vrednosti, tj. sumiranjem (po modulu 2) ispravnog izlaza sa sekvencom grešaka e, sekvencu grešaka moguće je posmatrati kao još jedan primarni ulaz u kompozivno logičko kolo sa verovatnoćom $p(e)$. Međutim, ako su otkazi vremenski zavisni i dodatno zavise i od ulaznih vrednosti *Parker-McCluskey* algoritam nije moguće direktno primeniti.

U opštem slučaju, ako se 2-ulazno logičko kolo nađe u stanju s_i, formiranom kao uređena $2M$-torka ulaznih vrednosti iz M sukcesivnih trenutaka, verovatnoća da izlaz logičkog kola bude jednak "1" iznosi

$$P_{out} = \sum_{i=1}^{N} \Pr\{s_i\} P_e(s_i) + \sum_{i=N+1}^{2^{2M}} \Pr\{s_i\}(1 - P_e(s_i)), \tag{2.11}$$

gde je $P_e(s_i)$ verovatnoća otkaza logičkog kola u stanju s_i, a stanja su indeksirana tako da prvih N dovodi izlaza pouzdanog logičkog kola jednakog "0". Verovatnoća pojave stanja s_i zavisi od verovatnoća pojave jedinice na ulazima u logičko kolo, koje se mogu označiti sa p_1 i p_2, respektivno. Tada verovatnoća pojave stanja s_i, sa $w(i)$ jedinica, pri čemu broj jedinica koje odogovaraju prvom, odnosno drugom, ulazu iznose $w_1(i)$ i $w_2(i)$, respektivno ($w(i) = w_1(i) + w_2(i)$), iznosi

$$\Pr\{s_i\} = p_1^{w_1(i)}(1 - p_1)^{M - w_1(i)} p_2^{w_2(i)}(1 - p_2)^{M - w_2(i)}. \tag{2.12}$$



Slika 2.10: Ilustracija metoda supstitucije promenljivih.

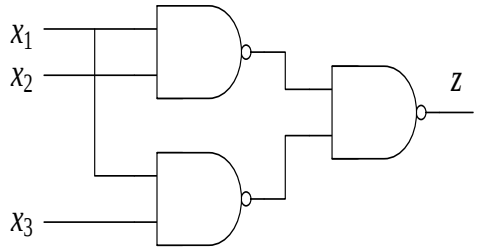
Treba napomenuti da je pretpostavljeno da su, za razliku od otkaza logičkog kola, ulazi logičkog kola vremenski nekorelisani. Jasno je da su stepeni u prethodnom izrazu posledica analize koja obuhvata više vremenskih trenutaka, a ne prostorne korelacije signala. Kako originalni *Parker-McCluskey* metod ne pravi razliku između ta dva slučaja, u ovom radu predložen je *metod supstitucije promenljivih*, koji razdvaja verovatnoće iz različitih vremenskih trenutaka. Tako se promenljive p_k $(k = 1, 2)$ zamenjuju sa M promenljivih $p_{k,j}$, $1 \leq j \leq M$, pa dobijamo

$$\Pr\{s_i\} = \prod_{j=1}^{w_1(i)} p_{1,j}^{w_1(i)} \prod_{j=w_1(i)+1}^{M} (1 - p_{1,j}) \prod_{j=1}^{w_2(i)} p_{2,j}^{w_2(i)} \prod_{j=w_2(i)+1}^{M} (1 - p_{2,j}). \tag{2.13}$$

Ako se supstitucija obavi za svaki primarni ulaz kroz sve moguće putanje stepeni u izlaznoj funkciji biće posledica isključivo prostorne korelacije. Pri tome se u svakom čvoru grafa logičkog kola supstitucija obavlja prema *roditelj-potomak* principu – u svakom nivou supstitucije "roditeljska" promenljiva se zamenjuje sa M promenljivih potomaka, kako je to ilustrovano na slici 2.10. Nakon što se verovatnoće primarnih ulaza propagiraju do primarnih izlaza i izvrše sve supstitucije, može se primeniti postupak eliminacije stepena opisan jednačinom (2.10), nakon čega se sve promenjive vraćaju u prvobitni oblik $(p_1, p_2, \ldots, p_m)$. Procena pouzdanosti logičkog kola može se dobiti poređenjem dobijene statistike i statistike njegovog pouzdanog parnjaka, prema nekom unapred zadatom kriterijumu.

Za potrebe verifikacije predloženog algoritma formiran je model grešaka koji se može opisati na sledeći način

$$P_e(s_i) = \Pr\{e = 1|s_i\} = A_i \Pr\{e = 1|s_0\} = A_i \varepsilon \;\; 1 \leq i \leq 2^{2M}, \tag{2.14}$$



Slika 2.11: Šema test kola T_1.

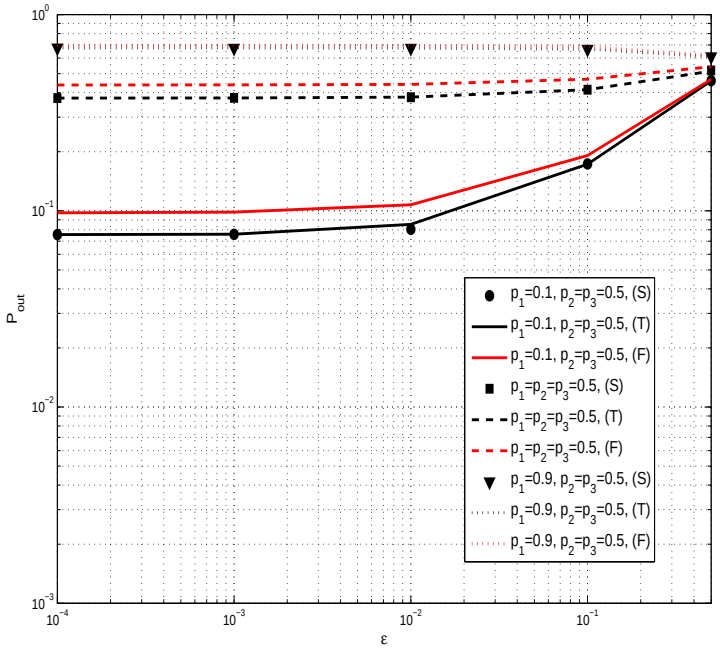
gde A_i koeficijent skaliranja koji se može odrediti kao

$$A_i = \frac{1}{p^{w(i)}}, \;\; p \geq 1. \tag{2.15}$$

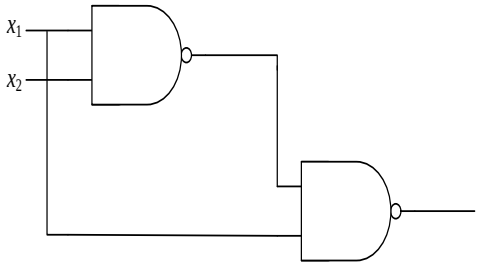
Ovaj model grešaka razmatran je u kontekstu uticaja šuma na pouzdanost logičkih kola u [61, 62] i pogodan je za verifikaciju predloženog algoritma, jer omogućava raznovrsnost u izboru verovatnoća otkaza u pojedinim stanjima. Verovatnoća da izlaz iz logičkog kola datog na slici 2.11 bude jednaka "1", u oznaci P_{out} predstavljena je na slici 2.12. Vrednosti dobijene predloženim algoritmom označene su sa (T), rezultati dobijeni Monte Karlo simulacijom nose oznaku (S), dok su sa (F) označene vrednosti koje bi se dobile kada se ne bi uračunala prostorna korelacija. Primetno je slaganje teorijski dobijenih vrednosti, sa rezultatima dobijenim simulacionim postupkom.

Treba primetiti da zbog potencijalnog postojanja asimetričnih putanja u šemi logičkih kola promenljive sa različitih nivoa supstitucije mogu postojati u finalnom izrazu. Da bi se efekat korelacije pravilno uzeo u obzir roditeljska promenljiva pomnožena sa potomcima takođe mora biti eliminisana. Tako, na primer, ako se u finalnom izrazu pojavi faktor $p_i, p_j, p_{i,1}, p_{i,11}$, potrebno ga je svesti na $p_i p_j$. Navedeni dodatak testiran je na primeru kola ilustrovanog na slici 2.13. Rezultati prezentovani na slici 2.14 ilustruju poklapanje rezultata dobijenih predloženim algoritmom sa rezultatima koje daje simulacioni postupak.

Kompleksnost predloženog rešenja zavisi od broja promenljivih potrebnih za simboličku analizu. Taj broj je u direktnoj vezi sa brojem primarnih ulaza i dužine putanja koje su podložne prostornoj korelaciji signala. Neka kolo ima m primarnih ulaza od kojih k stvaraju prostornu korelaciju, dok preostalih $m - k$ neće proizvesti stepene u finalnom izrazu i nema potrebe vršiti supstituciju njihovih promenljivih. Neka je sa N_i označen broj "korelisanih" putanja koje potiču od signala x_i. Neka je sa $D_i^{(j)}$ označen označena dužina j-te putanje ulaznog signala x_i.



Slika 2.12: Statistika izlaznog signala kola T_1 ($p = 2$).



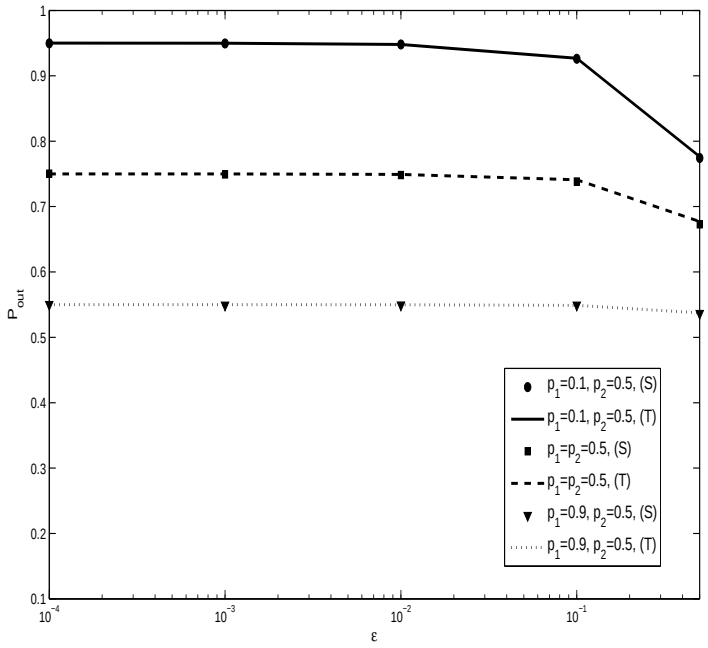
Slika 2.13: Šema test kola T_2.

Tada se ukupan broj potrebnih promenljivih može dobiti kao

$$N_{tot} = m - k + \sum_{i=1}^{k} \sum_{j=1}^{N_i} M^{D_i^{(j)}}. \tag{2.16}$$

Primećuje se da i za male vrednosti M, broj promenljivih koje treba analizirati može da bude veliki, ako u arhitekturi logičkog kola postoji veći broj korelisanih putanja. Simbolička analiza koja uključuje veći broj promenljivih je računarski zahtevna, pa predloženi algoritam nije praktičan za veća logička kola.

Pored algoritma *Parker-McCluskey* u literaturi je poznat i veći broj drugih metoda koji ne uzimaju u obzir korelisanu prirodu otkaza logičkih kola. Neki od najznačajnijih su, PDD (eng. *Probabilistic Decision Diagram*) [63], *Four-Event* [64], TPC (eng. *Trigonometric Probability Calculation*) [65], ili RALF (eng. *Reliability Analysis Logic Failures*) [66]. Za više informacija



Slika 2.14: Statistika izlaznog signala kola T_2 ($p = 2$).

o navedenim metodama čitalac se takođe upućuje i na monografsko izdanje [67]. Međutim, ni jednu od navedenih metoda nije moguće direktno proširiti tako da uzima u obzir vremensku zavisnost otkaza, pa oni nisu razmatrani u ovom radu. S druge strane, u literaturi su poznata i dva značajna algoritma koja dozvoljavaju da otkazi budu zavisni od ulaznih vrednosti, ali ne i od prethodnih trenutaka – PTM (eng. *Probabilistic Transfer Matrices*) model [57] i BN (eng. *Bayesian Network*) princip [68].

U PTM metodu logičko kolo se deli na poddomene, pri čemu se svakom podomenu dodeljuje PTM matrica. Poddomeni se dalje rastavljaju na elemente, tako da se razlikuju tri tipa elemenata i to *linija*, *razdelnik* (eng. *funout*) i *logičko kolo*. Smatra se da su linija i razdelnik pouzdani elementi, a da nepouzdanost potiče samo od logičkih kola. Svakom od elemenata dodeljuje se elementarna matrica verovatnoća koja definiše verovatnoću da pojedina ulazna kombinacija dovede do određene izlazne vrednosti. Pritom, vrste matrice verovatnoće odgovaraju ulaznim kombinacijama, a kolone izlaznim vrednostima. Tako su na primer matrice linije i razdelnika date redom,

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.17}$$

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.18}$$

S druge strane, matrica verovatnoća NAND logičkog kola ima sledeći oblik

$$\mathbf{G} = \begin{bmatrix} p_1 & 1-p_1 \\ 1-p_2 & p_2 \\ 1-p_3 & p_3 \\ 1-p_4 & p_4 \end{bmatrix}, \tag{2.19}$$

pri čemu vrste matrice $\mathbf{G}$ odgovaraju pojavi ulaznih kombinacija 00, 01, 10 i 11, od kojih zavise verovatnoće otkaza kola označene redom sa p_1, p_2, p_3 i p_4. Princip rada PTM biće ilustrovan na primeru kola datog na slici 2.15. Kompozitno logičko kolo se rastavlja na vertikalne domene M_1, M_2 i M_3, čije se PTM matrice računaju kao tenzorski proizvodi elementarnih matica elemenata obuhvaćenih poddomenima. Tako dobijamo

$$\mathbf{M}_1 = \mathbf{I} \otimes \mathbf{F} \otimes \mathbf{I}, \tag{2.20}$$

$$\mathbf{M}_2 = \mathbf{G} \otimes \mathbf{G}, \tag{2.21}$$

$$\mathbf{M}_3 = \mathbf{G}, \tag{2.22}$$

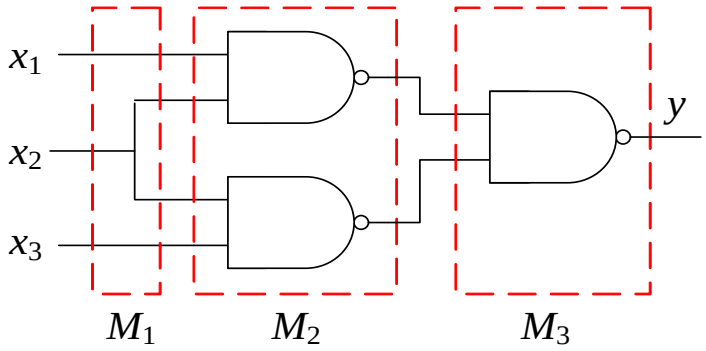
gde je sa $\otimes$ označen operator tenzorskog proizvoda dve matrice. PTM matrica kompletnog kola dobija se kao

$$\mathbf{P} = \mathbf{M}_1 \times \mathbf{M}_2 \times \mathbf{M}_3. \tag{2.23}$$

Ako je dat vektor verovatnoća svih mogućih ulaznih kombinacija $\mathbf{p}_x$ dimenzija 1×2^m (gde je m broj ulaza u kolo), vektor $\mathbf{p}_y = \mathbf{p}_x \times \mathbf{P}$ daće raspodelu vrednosti primarnih izlaza kola.

PTM metod moguće je modifikovati tako da opiše i vremensku zavisnost otkaza logičkih kola. Tada elementarne matrice verovatnoća treba proširiti tako da obuhvataju sve moguće ulazne kombinacije iz M susednih trenutaka i to

$$\mathbf{I}' = \left.\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ \cdots & \cdots \\ 0 & 1 \\ 0 & 1 \end{bmatrix}\right\} 2^M, \mathbf{F}' = \left.\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\right\} 2^M, \mathbf{G}' = \left.\begin{bmatrix} p_1 & 1-p_1 \\ p_2 & 1-p_2 \\ \cdots & \cdots \\ 1-p_{2^{2M-1}} & p_{2^{2M-1}} \\ 1-p_{2^{2M}} & p_{2^{2M}} \end{bmatrix}\right\} 2^{2M}, \tag{2.24}$$



Slika 2.15: Šema dekompozicije PTM algoritma.

gde su sa $\mathbf{I}'$,$\mathbf{F}'$ i $\mathbf{G}'$, označene modifikovane elementarne matrice linija, razdelnika i NAND logičkih kola. Kao i u originalnom algoritmu sada je moguće odrediti tenzorske proizvode $\mathbf{M}'_1$, $\mathbf{M}'_2$ i $\mathbf{M}'_3$. Međutim, u ovom slučaju nije moguće odrediti PTM matricu kompletnog kola. Umesto toga računaju se raspodele signala na izlazima pojedinih poddomena. Tako za zadati ulazni vektor $\mathbf{p}'_x$, dimenzija 1×2^{mM}, koji sadrži verovatnoće svih mogućih kombinacija ulaznih vrednosti iz M sukcesivnih trenutaka, računamo raspodelu izlaznih signala iz poddomena M_1 kao $\mathbf{p}_{I_1} = \mathbf{p}'_x \times \mathbf{M}'_1$. Uz pretpostavku da su verovatnoće pojava ulaznih vrednosti vremenski nekorelisane $\mathbf{p}_{I_1}$ se može iskoristiti za formiranje vektora koji bi sadržao verovatnoće pojave svih kombinacija izlaza iz M_1 u M sukcesivnih trenutaka. Množenjem tog vektora sa $\mathbf{M}_2$ dobija se raspodela izlaza iz M_2. Postupak se ponavlja dok se ne odredi raspodela primarnih izlaza.

Kompleksnost algoritama prezentovanih u ovom odeljku, u opštem slučaju, raste eksponencijalno sa povećanjem broja ulaza u kolo, pa su primenljivi samo za kola niskog nivoa kompleksnosti. Cilj izloženih rezultata nije konstrukcija efikasnih metoda za probabilističku analizu nepouzdanih logičkih kola, već više ilustracija kompleksnosti problema, za koji se pokazuje da je NP-kompletan. Praktično upotrebljivi algoritmi probabilističke analize morali bi da budu sub-optimalni i pojednostavljeni do nivoa kada ne bi predstavljli univerzalno rešenje. Korelacija internih signala u logičkom kolu koje bi predstavljalo neki iterativni dekoder kodova sa malom gustinom provera parnosti je visoka i postoji mogućnost da bi svaka aproksimacija značajno smanjila upotebljivost dobijenih rezultata. Zbog toga u nastavku rada probabilistički pristup na nivou logičkih kola nije dalje istraživan.

2.4 Zaključak

Problem nepouzdanosti elektronskih komponenti proizvdenih u CMOS tehnologiji je izražen i pristupa mu se na logičkom nivou. Rešenja koja obezbeđuju pouzdane logičke operacije kao i pouzdano memorisanje informacija na čipu postaju sve značajnija. Prvi korak je razumevanje izvora nepouzdanosti elektronskih sistema i adekvatno modelovanje otkaza. Iako su posledice koje izazivaju pojedini izvori otkaza poznate, nije ih moguće sasvim precizno uključiti u analizu različitih kombinacionih logičkih kola. Specifičnosti iterativnih dekodera kodova sa malom gustinom provera parnosti ogledaju se pre svega u veličini kao i velikoj međusobnoj korelaciji invernih signala, pa je izbor odgovarajućeg modela nepouzdanosti problem za sebe.

U ovom odeljku predložen je pojednostavljeni GOS pristup otkazima, koji nastaju kao posledica smanjenjog napajanja logičkih kola. Predloženi model opisuje korelisanu prirodu otkaza, kao i različite efekte logičkog maskiranja otkaza i, kako su to pokazali primeri simulacija, postiže prihvatljiv kompromis između jednostavnosti i preciznosti. Dodatno predložena su dva metoda probabilističke analize kombinacionih logičkih kola koji uzimaju u obzir vremensku zavisnost otkaza, što predstavlja generalniji pristup problemu od sličnih algoritama poznatih u literaturi.

Poglavlje 3

Osnove kodova sa malom gustinom provera parnosti

Rešavanje problema pouzdanog prenosa informacija korišćenjem zaštitinih kodova postalo je aktuelno u toku razvoja digitalnih komunikacija. Zaštitini kod podrazumeva preslikavanje informacione sekvence simbola u kodnu sekvencu koja će pored informacionog dela sadržati i redundansu, pri čemu je od interesa da količina redundanse bude što manja. Teorijski minimalne vrednosti redundanse (kolokvijalno nazvane kapacitet ili granica) za različite komunikacione kanale odredio je *Shannon* u svom fundamentalnom radu [9] iz 1948. godine. Kako *Shannon* nije predložio konkretno rešenje već samo postavio granice uspešnog komuniciranja, počela je decenijska potraga za konstrukcijom koda koji dostiže *Shannon*-ov kapacitet.

Prvi kodovi koji su se značajno približili *Shannon*-ovoj granici bili su turbo kodovi predloženi u članku [10]. Izuzetne korektivne sposobnosti *turbo kodova* praćene su velikom kompleksnošću, pa nisu predstavljali univerzalno rešenje problema pouzdanog komuniciranja. Paralelno sa razvitkom turbo kodova, istaživane su i sposobnosti LDPC kodova, koje je otkrio *Gallager* [26] 1963. godine, ali su u doba njihovog otkrića smatrani i suviše kompleksnim za praktične aplikacije. Pažnju na izuzetne teorijske osobine LDPC kodova skrenuo je *MacKay* [11] koji je iskoristio *Tanner*-ovo zapaženje da se LDPC kodovi mogu predstaviti grafom [69], kada počinje intezivan razvoj ove oblasti. Da se LDPC kodovi mogu približiti *Shannon*-ovoj granici na samo deseti deo decibela, prvi je pokazao *Richardson* [32]. Iako je kod koji je konstruisao imao dužinu 10^6 bita, nije poznata ni jedna druga klasa kodova koja bi prevazišla *Richardson*-ov LDPC kod. Posebano je značajno što kompleksnost iterativnih dekodera LDPC kodova raste samo linearno sa dužinom koda, što je osobina koju nema većina drugih klasa kodova.

Raznovrsnosti u izboru dužine, kodnog količnika, kao i algoritama dekodovanja čine LDPC kodove atraktivnim za praktične primene. Značaj LDPC kodova poslednjih godina prevazišao je akademske okvire, tako da su se našli u predlogu nekoliko značajnih telekomunikacionih protokola kao što su DVB-S2, IEEE 802.3an – 10GBASE-T, IEEE 802.11n (WiFi) i IEEE 802.16e (WiMax). Primenu su našli i u različitim uređajima za skladištenje informacija.

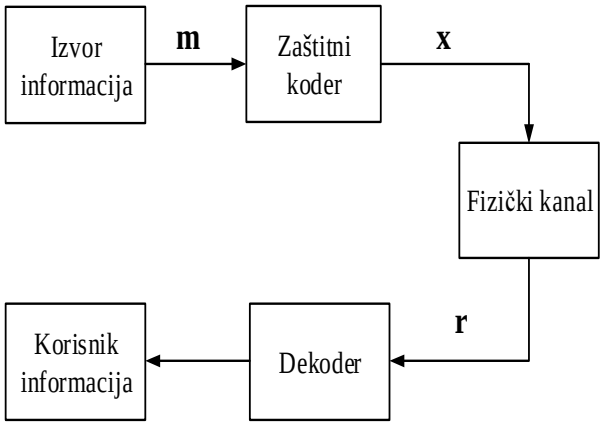
Ostatak poglavlja organizovan je na sledeći način. U Odeljku 3.1 izloženi su osnovni pojmovi vezani za LDPC kodove i kodove konstruisane nad grafovima. U Odeljaku 3.2 ukratko su opisani najznačajniji metodi konstrukcije LDPC kodova i objašnjeno kako se ovi kodovi mogu efikasno kodovati. Odeljak 3.3 posvećen je iterativnim dekoderima LDPC kodova, sa posebnim osvrtom na fenomene koji negativno utiču na uspešnost dekodovanja. U Odeljku 3.4 opisana je klasa LDPC kodova koja ostvaruje izuzetne teorijske karakteristike, dok se u Odeljku 3.5 može naći opis metode koja se često koristi pri asimptotskoj analizi iterativnih dekodera.

3.1 Osnovni pojmovi

Linearni blok kod transformiše blok $\mathbf{m} = (m_1, m_2, \ldots, m_k)$ informacionih simbola u kodnu reč $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $n > k$, pri čemu se kodni količnik koda definiše kao $R = k/n$. Ako skup svih 2^k kodnih reči sačinjavaju potprostor vektorskog prostora od ukupno 2^n simbolskih sekvenci (dužine n), tada preslikavanje kojim se formiraju kodne reči nazivamo *linearnim blok kodom*. U ovom radu posmatrani su samo binarni kodovi, pa će se kodni simboli nazivati bitima. Linearni blok kodovi opisuju se *generišućom matricom* $\mathbf{G} = [g_{i,j}]$, $(1 \leq i \leq k, 1 \leq j \leq n)$ dimeniza $k \times n$ koja predstavlja bazis vektorskog potprostora, pri čemu se proces formiranja kodnih reči može predstaviti kao

$$\mathbf{x} = \mathbf{m} \cdot \mathbf{G}, \tag{3.1}$$

gde se operacije obavljaju u *Galois*-ovom polju GF(2). Analogno, poznajući osobine vektorskih potprostora, za svaku matricu $\mathbf{G}$ postoji ortogonalna matrica $\mathbf{H} = [h_{i,j}]$, dimenzija $n - k \times n$ tako da važi $\mathbf{G} \cdot \mathbf{H}^T = 0$, gde je T operator transpozicije matrice. Matrica $\mathbf{H}$ se naziva *kontrolnom matricom* ili *matricom provera parnosti* (eng. *parity-check matrix*). Kako kontrolna matrica takođe jednoznačno opisuje linearni blok kod, to ju je moguće iskoristiti u procesu dekodovanja. Tako, ako je primljena sekvenca $\mathbf{r} = (r_1, r_2, \ldots, r_n)$ jednostavan dekoder može računati samo *sindrom* $\mathbf{s} = (s_1, s_2, \ldots, s_{n-k})$, kao $\mathbf{s} = \mathbf{r} \cdot \mathbf{H}^T$, pri čemu ako je supp$(s) = 0$



Slika 3.1: Pojednostavljeni *Shannon*-ov dijagram komuniciranja.

smatra se da je primljena sekvenca kodne reč. U suprotnom moguće je izvršiti neku dodatnu operaciju ispravljanja grešaka ili vršiti retransmisiju primljene sekvence.

Od interesa za analizu linearnih blok kodova su i *težine vrsta* i *kolona* kontrolne matrice. Tako težina i-te vrste kontrolne matrice u oznaci ρ_i predstavlja broj jedinica u i-toj vrsti matrice $\mathbf{H}$, tj.

$$\rho_i := \text{supp}([h_{i,1}, h_{i,2}, \ldots, h_{i,n}]), \tag{3.2}$$

dok je, slično, težinu j-te kolone kontrolne matrice, γ_j, moguće definisati kao

$$\gamma_j := \text{supp}([h_{1,j}, h_{2,j}, \ldots, h_{j,n-k}]^T). \tag{3.3}$$

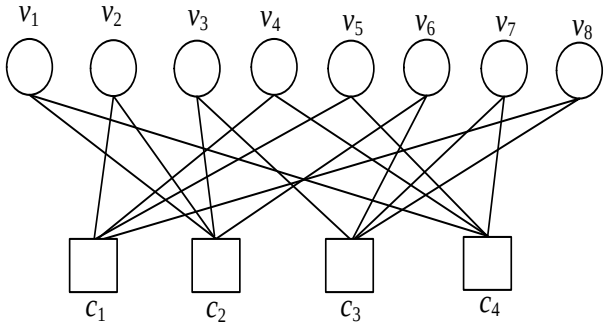
Uobičajeno je proces ispravljanja grešaka posmatrati u okviru *Shannon*-ovog dijagrama komuniciranja, čija je pojednostavljena verzija data na slici 3.1. Nepozdanost kanala komuniciranja manifestuje se greškama u sekvenci koja napušta komunikacioni kanal. Izlazna sekvenca iz kanala ne mora nužno biti binara, tj. kanal može pružiti i *meke* informacije, koje je moguće koristiti za povećanje pouzdanosti sistema za prenos. Kako je problem komuniciranja posmatran u ovom radu uglavnom vezan za primenu u memorijskim sistemima, gde meke informacije najčešće nije moguće prikupiti, to će biti smatrano da je kanal binaran i simetričan (eng. *Binary Symmetric Channel*, BSC), tako da je verovatnoća invertovanja ista za svaki bit.

Kodovi sa malom gustinom provera parnosti predstavljaju specijalni slučaj linearnih blok kodova čija je kontrolna matrica $\mathbf{H}$ "male gustine", tj. ima mali broj jedinica u odnosu na dimenzije matrice (eng. *sparse matrix*). Iako mala gustina provera parnosti čini LDPC kodove efikasnim za hardverske implementacije, šezdesetih godina prošlog veka, kada je *Gallager* [26] predložio ove kodove, smatrali su se i suviše kompleksnim za praktičnu primenu. Nakon

perioda zapostavljenosti, reprezentacija LDPC kodova grafovima posebno je značajna za dalji razvoj ove oblasti. Graf u oznaci $G = (U, E)$ definiše se na osnovu skupa čvorova U i skupa grana E. Grana $e \in E$ povezuje dva čvora $u_1 \in U$ i $u_2 \in U$, u oznaci $e = (u_1, u_2)$. Čvorovi u_1 i u_2 nazivaju se susedima. Kardinalni broj skupa U, u oznaci $|U|$ naziva se *redom grafa*, dok je $|E|$ *veličina grafa*. Skup suseda čvora u označava se sa $\mathcal{N}(u)$, dok je skup grana povezanih na čvor u $\mathcal{E}(u)$. *Stepen* čvora u označava se sa $|\mathcal{N}(u)|$, dok se prosečni stepen grafa definiše kao $\bar{d} := 2|E|/|U|$. Svaki graf koji ima konačan broj čvorova sadrži cikluse, pri čemu se dužina najkraćeg siklusa naziva $girth$-om i označava sa g.

Bipartitni graf, označen kao $G = (V \cup C, E)$ sastavljen je od dva komplementarna skupa čvorova V i C, $V \cap C = \emptyset$, takvih da za svaki čvor $\forall v \in V$ važi $\mathcal{N}(v) \subset C$, i obrnuto $\forall c \in C$ važi $\mathcal{N}(c) \subset V$. Drugim rečima svi susedi nekog čvora moraju biti iz komplementarnog potskupa. Bipartitni graf kod koga $\forall v \in V$ $|\mathcal{N}(v)| = \gamma$, $\gamma > 0$ naziva se γ-levo-regularnim, dok ako je $\forall v \in C$ $|\mathcal{N}(c)| = \rho$, $\rho > 0$, naziva se ρ-desno-regularnim. Kontrolna matrica LDPC koda $\mathbf{H}$ je *matrica susedstva* (eng. *adjacency matrix*) bipartitnog grafa G, koji se tada naziva *Tanner*-ovim grafom [69]. Kolone matrice $\mathbf{H}$ odgovaraju čvorovima iz skupa V, kada se oni nazivaju *varijabilnim* ili *simbolskim* čvorovima. Slično, vrste kontrolne matrice LDPC koda odgovaraju čvorovima iz skupa C kada se oni nazivaju *kontrolnim*. Postojanje jedinica u kontrolnoj matrici definiše susedstvo čvorova grafa, tako da u *Tanner*-ovom grafu postoji grana $e = (v, c)$ ako i samo ako važi $h_{c,v} = 1$. Težine vrsta i kolona kontrolne matrice odgovaraju stepenima bipartitnog grafa tako da je $|\mathcal{N}(c)| = \rho_c, \forall c \in C$ i $|\mathcal{N}(v)| = \gamma_v, \forall v \in V$. Od interesa je definisati i stablo grafa dubine ℓ, čiji je koren u čvoru $v \in V$, u oznaci $\mathcal{T}_v^{(\ell)}(\mathcal{V}_v^{(\ell)} \cup \mathcal{C}_v^{(\ell)}, \mathcal{E}_v^{(\ell)})$.

Ako su težine svih kolona i težine kontrolne matrice iste i iznose γ i ρ, respektivno, govori se o (γ, ρ)-regularnim kodovima. U suprotnom, za LDPC kod se kaže da je *iregularan*. Kako je $|V| = n$ ukupan broj grana u γ-levo-regularnom *Tanner*-ovom grafu iznosi $|\mathcal{E}| = n\gamma$. S druge strane, treba napomenuti da kontrolna matrica LDPC koda ne mora imati pun rang, odnosno broj vrsta može biti veći od $n - k$. Broj vrsta kontrolne matrice iznosi $|C| = \gamma n/\rho$. Tada se govori o *konstrukcionom kodnom količniku* koji iznosi $1 - \gamma/\rho$ dok se za stvarni kodni količnik može jedino tvrditi $R \geq 1 - \gamma/\rho$. Stvarni kodni količnik moguće je otkriti procenom ranga kontrolne matrice, na primer *Gauss*-ovim metodom eliminacije.



Slika 3.2: *Tanner*-og graf koda datog matricom (3.4).

Neka je data kontrolna matrica LDPC konstrusana

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{3.4}$$

koja se može predstaviti *Tanner*-ovim grafom čiji su čvorovi $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$ i $C = \{c_1, c_2, c_3, c_4\}$. Veze između čvorova ilustrovane su na slici 3.2. Pritom treba zapaziti da je reč o $(2, 4)$-regularnom kodu. U najopštijem zapisu varijabilnim čvorovima se mogu pridružiti primljeni kodni biti tako da važi $v_i = r_i$, $1 \leq i \leq n$. Slično, kontrolnim čvorovima se mogu pridružiti vrednosti sindroma, pa tako u našem primeru imamo

$$\begin{aligned} c_1 &= r_2 \oplus r_4 \oplus r_5 \oplus r_8 \\ c_2 &= r_1 \oplus r_2 \oplus r_3 \oplus r_6 \\ c_3 &= r_3 \oplus r_6 \oplus r_7 \oplus r_8 \\ c_4 &= r_1 \oplus r_4 \oplus r_5 \oplus r_7. \end{aligned} \tag{3.5}$$

S druge strane, moguće je različite poruke pridružiti granama *Tanner*-ovog grafa, što će biti detaljnije analizirano u odeljku posvećenom algoritmima dekodovanja.

Treba naglasiti da predstava LDPC kodova *Tanner*-ovim grafom nije jedina moguća. Poznato je da se kontrolna matrica $\mathbf{H}$ može predstavti *hipergrafom* tako da grane hipergrafa odgovaraju varijabilnim čvorovima, dok su čvorovi kojima su povezane grane kontrolni [70]. U ovom radu dominantno će biti korišćena predstava *Tanner*-ovim grafom, dok se alternativna predstava više koristi pri analizi *generalizovanih LDPC kodova*, gde varijabilni čvorovi susedi nekog kontrolnog čvora predstavljaju kodne bite kraćeg unutrašnjeg linearnog blok koda. Više informacija o generalizovanim LDPC kodovima moguće je pronaći u člancima [71–73].

3.2 Konstrukcija i kodovanje LPDC kodova

Kako se LDPC kodovi definišu preko kontrolne (a ne generišuće) matrice, to se i njihova konstrukcija svodi na pronalaženje kontrolne matrice koja ispunjava neke unapred zadate kriterijume. Kriterijumi su najčešće vezani za dužinu koda, kodni količnik, težine vrsta i kolona, ili vrednost *girth*-a. U nastavku će ukratko biti izloženi najznačajniji principi konstrukcije struktuiranih LDPC kodova. Kako se po pravilu kodovi konstruišu u nesistematskoj formi, to proces kodovanja nije trivijalan. Jedan mogući princip, koji su predložili *Richardson* i *Urbanke* [74], dat je na kraju odeljka.

3.2.1 Kodovi na bazi konačnih geometrija

U ovom delu odeljka biće ukratko date osnove kodova na bazi konačnih gemetrija, bez navođenja osnovnih osobina ovih geometrija. Više informacija o toj temi moguće je pronaći u [75]. Da se LDPC kodovi mogu konstruisati na osnovu tačaka i linija konačnih geometrija prvi su uočili *Kou*, *Lin* i *Fossorier* u sada već klasičnom članku [76]. Neka je $\mathbf{Q}$ konačna geometrija sačinjena od n tačaka i J linija sa sledećim fundamentalnim osobinama:

1) svaka linija se sastoji od ρ tačaka;

2) svaka tačka leži na tačno γ linija;

3) dve tačke su povezane jednom i samo jednom linijom;

4) dve linije se mimoilaze ili se seku u jednoj i samo jednoj tački.

Neka su tačke konačne geomerije $\mathbf{Q}$ date sa skupom tačaka $\{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$. Slično, neka je $\mathbf{L}$ linija u geometriji $\mathbf{Q}$. Tada se može formirati binarni vektor $\mathbf{x_L} = (x_1, x_2, \ldots, x_n)$, pri čemu važi

$$v_i = \begin{cases} 1, & \text{ako se } \mathbf{p}_i \text{ nalazi na liniji } \mathbf{L} \\ 0, & \text{inače.} \end{cases} \tag{3.6}$$

Formiranje prethodno opisanih vektora za svaku od J linija konačne geometrije generiše matricu $\mathbf{H}$ dimenzija $J \times n$ sa sledećim osobinama: 1) svaka kolona ima tačno ρ jedinica; 2) svaka kolona ima tačno γ jedinica; 3) ne postoje dve vrste sa više od jedne jedinice na istim pozicijama; 4) ne postoje dve kolone sa više od jedne jedinice na istim pozicijama. Primetiti da su

kodovi na bazi konačnih geometrija regularni, a da je dužina najkraćeg ciklusa $g > 4$. Poslednja činjenica je važna, jer obezbeđuje da kodovi imaju γ ortoganalnih provera parnosti na svaki od n kodnih bita, a minimalno *Hamming*-ovo rastoje u kodu je bar $\gamma + 1$. Kodove je moguće konstruisati na osnovu euklidske (EG), projektivne (PG) ili afine geometrije (AG). Euklidska geometrija EG$(m, 2^s)$ sadrži $n = 2^{ms}$ tačaka, pri čemu je svaka tačka predstavljena m-torkom čiji su elementi iz polja GF(2^m). EG kodovi se najčešće formiraju izbacivanjem tačke svih nula, kao i svih linija na kojima leži ova tačka. Tada kod postaje cikličan, što pojednostavljuje proces kodovanja. Ukupan broj linija koji preostaje sačinjava skup $\mathcal{L}$ [76]

$$|\mathcal{L}| = \frac{(2^{(m-1)s} - 1)(2^{ms} - 1)}{2^s - 1}. \tag{3.7}$$

Svaka tačka u EG$(m, 2^s)$ sadržana je u tačno

$$\gamma = \frac{2^{ms} - 1}{2^s - 1} - 1, \tag{3.8}$$

linija iz $\mathcal{L}$. Kako svaka linija sadrži tačno 2^s tačaka, to je $\rho = 2^s$. Kako je minmalno *Hamming*-ovo rastojanje EG koda $d_{min} \geq \gamma + 1$ to je ovim kodom moguće ispraviti bar $t = \lfloor((2^{ms} - 1)/(2^s - 1) - 1)/2\rfloor$ grešaka dekoderom na bazi većinskog odlučivanja, koji će detaljno biti analiziran u Poglavlju 4. Parametri EG kodava kada je $m = 2$ dati su u tabeli nastavku.

Tabela 3.1: Parametri EG$(2, 2^s)$ kodova.

Dužina	$n = 2^{2s} - 1$
Broj informacionih bita	$k = 3^s - 1$
Broj kontrolnih bita	$n - k = 2^{2s} - 3^s$
Minimalno rastojanje u kodu	$2^s + 1$
Težina vrste	$\gamma = 2^s$
Težina kolone	$\rho = 2^s$

LDPC kodove moguće je konstruisati i na bazi projektivne geometrije. Neka je α primitivni element *Galois*-ovog polja GF$(2^{(m+1)s})$. Tada su elementi $\alpha^0, \alpha^1, \ldots, \alpha^n$, gde je $n = (2^{(m+1)s} - 1)/(2^s - 1)$, tačke m-dimenzione projektivne geometrije PG$(m, 2^s)$. Strukturni parametri PG kodova, kada je $m = 2$, dati su u tabeli u nastavku. Treba pomenuti i kodove bazirane na afinoj geometriji, AG$(2, 2^s)$, sa konstrukcionim parametrima $\gamma = 2^s$, $\rho = 2^s + 1$. Minimalno rastojanje AG kodova iznosi $d_{min} = 2^s + 2$.

Tabela 3.2: Parametri PG$(2, 2^s)$ kodova.

Dužina	$n = 2^{2s} + 2^s + 1$
Broj informacionih bita	$k = 2^{2s} + 2^s - 3^s$
Broj kontrolnih bita	$n - k = 3^s + 1$
Minimalno rastojanje u kodu	$2^s + 2$
Težina vrste	$\gamma = (2^{ms} - 1)/(2^s - 1)$
Težina kolone	$\rho = (2^{ms} - 1)/(2^s - 1)$

Kodovi bazirani na konačnim geometrijama ostvaruju izuzetne performanse i mogu se relativno jednostavno dekodovati dekoderom na bazi većinskog odlučivanja, kada je broj grešaka koje se tako mogu ispraviti blizak korektivnoj sposobnosti dekodera maksimalne verodostojnosti (eng. *Maximum Likelihood*, ML). Međutim, iako su kodni količnici ovakvih kodova visoki, broj kodova koji se mogu konstruisati relativno je mali. Takođe, broj jedinica u kontrolnoj matrici raste eksponencijalno sa povećanjem dužine koda, što često dovodi do neprihvatljivog nivoa kompleksnosti kodova većih dužina.

3.2.2 Kvazi ciklični LDPC kodovi

Veću slobodu u izboru parametara koda pruža konstrukcioni princip koji je predložio *Tanner* [4]. Ovaj princip generiše kvazi-ciklične (eng. *Quasi-Cyclic*, QC) LDPC kodove formirane nad *Galois*-ovim poljem GF(m), gde je m prost broj. Nenulti elementi iz GF(m) formiraju multiplikativnu grupu. Neka su a i b dva elementa iz multiplikativne grupe čiji su redovi [1] $o(a) = \rho$ i $o(b) = \gamma$, respektivno. Zatim se definiše permutaciona matrica $\mathbf{P} = [p_{s,t}]$, dimenzija $j \times k$, čiji su elementi iz GF(m) određeni kao $p_{s,t} = b^s a^t$, pa imamo

$$\mathbf{P} = \begin{bmatrix} 1 & a & a^2 & \cdots & a^{\rho-1} \\ b & ab & a^2b & \cdots & a^{\rho-1}b \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ b^{\gamma-1} & ab^{\gamma-1} & a^2b^{\gamma-1} & \cdots & a^{\rho-1}b^{\gamma-1} \end{bmatrix} \tag{3.9}$$

[1] red elementa grupe a iz GF(m), $o(a)$, definiše se kao najmanji prirodan broj ρ za koji važi $a^\rho \equiv 1 (\mathrm{mod} m)$

Kontrolna matrica LDPC koda konstriše se kao niz submatrica cirkulnata, dimenzija $j \times k$, kao

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_1 & \mathbf{I}_a & \mathbf{I}_{a^2} & \cdots & \mathbf{I}_{a^{\rho-1}} \\ \mathbf{I}_b & \mathbf{I}_{ab} & \mathbf{I}_{a^2 b} & \cdots & \mathbf{I}_{a^{\rho-1} b} \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ \mathbf{I}_{b^{\gamma-1}} & \mathbf{I}_{ab^{\gamma-1}} & \mathbf{I}_{a^2 b^{\gamma-1}} & \cdots & \mathbf{I}_{a^{\rho-1} b^{\gamma-1}} \end{bmatrix} \tag{3.10}$$

gde je $\mathbf{I}_x$, jedinična matrica, dimenzija $m \times m$, čije su vrste ciklično pomerene ulevo za x pozicija. Jasno je da cirkulant na poziciji (s, t) odgovara pomeraju vrsta jedinične matrice za $p_{s,t}$ pozicija ulevo. Svaka vrsta matrice $\mathbf{H}$ sadrži tačno ρ jedinica, dok je broj jedinica u kolonama jednak γ, pa govorimo o (γ, ρ)-regularnim kodovima. Pogodnim izborom elemenata a i b, kao i parametra m moguće je konstruisati kodove različitih dužina i kodnih količnika, od kojih su neki dati u tabeli u nastavku. Od posebnog interesa je činjenica da se ovom metodom

Tabela 3.3: Pregled kvazi-cikličnih kodova [4].

n	γ	ρ	R	m
21	2	3	0,3809	7
93	2	3	0,3441	31
129	2	3	0,3441	43
155	3	5	0,4	31
305	3	5	0,4	61
755	3	5	0,4	151
905	3	5	0,4	181
1205	3	5	0,4	241
2041	3	5	0,7702	157
3641	5	11	0,5465	331
5219	3	17	0,8239	307
11555	3	5	0,4001	2311

mogu kontruisati kodovi sa malom težinom kolona kontrolne matrice. Kodovi kod kojih je $\gamma = 3$ ili $\gamma = 4$ smatraju se praktično značajnijim zbog niske kompleksnosti dekodovanja. Tako se, na primer, (3,5)-regularni kod dužine $n = 155$ i kodnog količnika $R = 0,4$, u oznaci QC(155,64), smatra referentnim i često se performanse različitih dekodera ispituju na ovom kodu. U literaturi se kod još naziva i *Tanner*-ovim kodom, dok njegova kontrolna matrica ima

oblik

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_1 & \mathbf{I}_2 & \mathbf{I}_4 & \mathbf{I}_8 & \mathbf{I}_{16} \\ \mathbf{I}_5 & \mathbf{I}_{10} & \mathbf{I}_{20} & \mathbf{I}_9 & \mathbf{I}_{18} \\ \mathbf{I}_{25} & \mathbf{I}_{19} & \mathbf{I}_7 & \mathbf{I}_{14} & \mathbf{I}_{28} \end{bmatrix} \tag{3.11}$$

Minimalno *Hamming*-ovo rastojanje *Tanner*-ovog koda iznosi $d_{min} = 20$. Iako je teoretski ML dekoderom moguće ispravljanje t grešaka za svako $t \leq \lfloor (d_{min} - 1)/2 \rfloor = 9$ grešaka, do sada nije konstruisan praktičan dekoder koji garantuje ispravljanje svih četvorostrukih grešaka na ovom kodu. Generalno, mane QC kodova izražene su u regionu sa niskom verovatnoćom greške u kanalu (eng. *error-floor*), kada na performanse koda utiču greške male težine. Uzrok ovakvog ponašanja QC kodova vezan je za postojanje struktura u *Tanner*-ovom grafu ovih kodova, zvanih *trapping set*-ovi o kojima će više reči biti u narednim odeljcima. Prednost upotrebe QC kodova ogleda se u njihovoj kvazi-cikličnoj strukturi, što pojednostavljuje implementaciju kodera i dekodera. Koder se na primer može implementirati koristeći pomeračke registre, dok se dekodovanje može podeliti na subdomene (slojeve) (eng. *layers*).

3.2.3 PEG-LDPC i LS-LDPC kodovi

Algebarsku konstrukciju LPDC kodova, u kojoj se direktno generiše *Tanner*-ov graf iterativnim postupkom, nazvana PEG (eng. *Progressive Edge-Growth*) predložio je *Hu* u popularnom članku [77]. Pri tome se u svakoj iteraciji konstrukcionog algoritma dodaje po jedna grana grafa, dok se ne dođe do unapred zadatog broja grana, definisanog preko dimenzija kontrolne matrice i težina vrsta i kolona matrice. Neka je $\mathcal{E}'(u)$ skup grana povezanih na čvor u u trenutnoj konstelaciji grafa. PEG princip konstrukcije može se sumirati koracima datim u nastavku koji se ponavljaju dok se sve grane ne unesu na graf.

- Ako u trenutnoj konstelaciji grafa za neke čvorove $v \in V$ i $c \in C$ važi $\mathcal{E}'(v) = \emptyset$ i $\mathcal{E}'(c) \leq \mathcal{E}'(c'), \forall c' \in C$ povezati čvorove v i c granom $e = (v, c)$.

- Ako za neki čvor $v \in V$ važi $\mathcal{E}'(v) \neq \emptyset$, posmatra se stablo do dubine ℓ izabrano tako da (i) $\mathcal{C}_v^{(\ell)}$ prestane da se povećava a maksimalna dubina stabla nije dosegnuta, ili (ii) da važi $C \setminus \mathcal{C}_v^{(\ell)} \neq \emptyset$ i $C \setminus \mathcal{C}_v^{(\ell+1)} = \emptyset$. U graf se dodaje grana $e = (v, c)$, pri čemu se bira $c \in C \setminus \mathcal{C}_v^{(\ell)}$ takvo da je $\mathcal{E}'(c) \leq \mathcal{E}'(c'), \forall c' \in C \setminus \mathcal{C}_v^{(\ell)}$.

Hu je takođe pokazao da minimalno *Hamming*-ovo rastojanje ($\gamma \geq 3, \rho$)-regularnih PEG kodova zadovoljava [77]

$$d_{min} \geq 1 + \frac{\gamma((\gamma-1)^{\lfloor (g-2)/4 \rfloor}) - 1)}{\gamma - 2}, \tag{3.12}$$

gde za vrednost *girth*-a g važi

$$g \geq 2\Big(\lfloor \frac{\log\left(n(\gamma - \gamma/\rho - 1) + 1\right)}{\log(\gamma-1)(\rho-1)} - 1 \rfloor + 2\Big). \tag{3.13}$$

Primetiti da se za razliku od QC ili kodova na bazi konačnih geometrija, PEG kodovi direktno konstruišu tako da zadovoljavaju uslov vezan za *girth Tanner*-ovog grafa. Povećanje *girth*-a direktno unapređuje osobine kodova u *error-floor* regionu, pa PEG kodovi najčešće ne sadrže male *trapping set*-ove. U linerature se posebno navodi (3,6)-regularni kod označen sa PEG(502,252), dužine $n = 502$, kodnog količnika $R = 252/502 \approx 0,5$, i *girth*-a $g = 8$. Iako je proces kodovanja PEG kodova linearna funkcija dužine koda n, kvazi-cikličnost ovih kodova nije pokazana, pa samim tim ni efikasna implementacija kodera i dekodera.

S druge strane, *Nguyen* [78] je pokazao kako se jednostavnost QC-LDPC kodova i dobre performanse PEG-LDPC kodova u *error-floor* regionu mogu iskombinovati u jednoj klasi kodova baziranih na latinskim kvadratima (eng. *Latin Squares*, LS). Latinski kvadrat $\mathcal{L} = [\ell_{i,j}]$, dimenzija $q \times q$, je matrica koja sadrži elemente iz *Galois*-ovog polja $GF(q)$, pri čemu se svaki simbol pojavljuje tačno jednom u svakoj vrsti i koloni. Može se izabrati matrica $\mathbf{W} = [w_{i,j}]$, dimenzija $\gamma \times \rho$, čiji su elementi iz polja $GF(q)$. Tada je kontrolnu matricu LS-LDPC kodova moguće formirati na sledeći način

$$\mathbf{H} = \begin{bmatrix} f(w_{1,1}) & f(w_{1,2}) & f(w_{1,3}) & \cdots & f(w_{1,\rho}) \\ f(w_{2,1}) & f(w_{2,2}) & f(w_{2,3}) & \cdots & f(w_{2,\rho}) \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ f(w_{\gamma,1}) & f(w_{\gamma,2}) & f(w_{\gamma,3}) & \cdots & f(w_{\gamma,\rho}) \end{bmatrix}, \tag{3.14}$$

gde se mapiranje f: $\alpha \in \mathrm{GF}(q) \rightarrow \mathbf{M} = [m_{i,j}]_{q \times q}$ definiše kao $f(\alpha) = \mathbf{M}$, gde je

$$m_{i,j} = \begin{cases} 1, & \text{ako je } \ell_{i,j} = \alpha \\ 0, & \text{inače.} \end{cases} \tag{3.15}$$

Može se uočiti da su dimenzije kontrolne matrice $q\gamma \times q\rho$, dok su γ i ρ kao i ranije težine kolona i vrsta kontrolne matrice, respektivno. Sloboda pri izboru matrice $\mathbf{W}$ omogućava konstrukciju velikog broja kodova različitih dužina i performansi. Izuzetnost *Nguyen*-ov rada ogleda se

u činjenici da je uspeo da izdvoji kodove čiji *Tanner*-ovi grafovi ne sadrže strukture koje su zaslužne za neuspešno ispravljanje grešaka male težine korišćenjem iterativnih dekodera (*trapping set*-ovi). Tako na primer, permutaciona matrica

$$\mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \alpha^5 & \alpha^{15} & \alpha^{23} \\ 0 & \alpha^{16} & \alpha^4 & \alpha^{24} & \alpha^{12} \end{bmatrix}, \tag{3.16}$$

gde je α primitivni element polja GF(31), dovodi do formiranja koda označenog sa LS(155,64) koji ima iste konstrukcione parametre kao kod QC(155,64), ali ne sadrži *trapping set*-ove male težine. Međutim, sličan zaključak nije moguće uspostaviti za znatan broj drugih QC kodova.

Treba napomenuti i druge istorijski značajne klase kodova koje nisu opisane u prethodnim poglavljima. Tako se dugo smatralo da nestruktuirani (pseudoslučajni) kodovi koje su predložili *Gallager* [26] i *MacKay* [11, 79, 80] ostvaruju superiorne performanse, u odnosu na algebarski konstruisane kodove. Međutim, kasniji radovi opovrgli su taj zaključak, pa se i kodovi konstrisani na osnovu balansiranog nekompletnog blok dizajna (eng. *balanced incomplete block design*) [81–84] smatraju praktično značajnim. S druge strane, *Rosenthal* i *Vontobel* [85] su pokazali kako se značajni LDPC kodovi mogu konstrisati na osnovu *Ramanujan*-ovih grafova, što je originalno primetio i *Margulis* [86, 87].

3.2.4 Kodovanje LDPC kodova

Transfmisanje kontrolne matrice $\mathbf{H}$ u sistematsku formu nije jednostavan zadatak. Tako je kompleksnost kodovanja u opštem slučaju $O(n^2)$, dok se kompleksnost dekodera može iskazati kao $O(n)$, kako su to primetili *Richardson* i *Urbanke* u referentnom članku [74]. Autori su predložili algoritam koji polazeći od kontrolnih matrica LDPC kodova predstavljenih u prethodnim izlaganjima, formira sistematsku formu kodnih reči $\mathbf{x} = [\mathbf{m}\ \mathbf{p}_1\ \mathbf{p}_2]$, gde je sa $\mathbf{m}$ označen informacioni sadržaj koji se koduje, dok $\mathbf{p}_1$ i $\mathbf{p}_2$ predstavljaju bite provera parnosti.

Algortam se sastoji od dva dela i to (i) inicijalnog (eng. *preprocessing*) dela i (ii) dela u kome se obavlja kodovanje. U toku incijalnog dela kontrolnu matricu $\mathbf{H}$ potrebno je prikazati kao *približnu donju trougaonu matricu*, tj. u formi

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{T} \\ \mathbf{C} & \mathbf{D} & \mathbf{E} \end{bmatrix}, \tag{3.17}$$

gde su $\mathbf{A}$,$(\gamma/\rho n-a)\times n(1-\gamma\rho)$, $\mathbf{B}$, $(\gamma/\rho n-a)\times a$, $\mathbf{C}$, $(a\times n(1-\gamma\rho)$, $\mathbf{D}$, $a\times a$ i $\mathbf{E}$, $a\times(\gamma/\rho n)$ *sparse* matrice, dok je $\mathbf{T}$, $(\gamma\rho n-a)\times(\gamma\rho n-a)$ donja trougaona matrica. Parametar a naziva se *raskorakom* (eng. *gap*) i njegova vrednost zavisi od dužine koda. Dekompoziciju predstavljenu prethodnom relacijom moguće je izvršiti "pohlepnim" (eng. *greedy*) algoritmom, tako da vrednost a bude što je moguće manja, što je detaljno predstavljeno u članku [74]. Takođe, vrednost raskoraka a određuje dužinu delova $\mathbf{p}_1$ i $\mathbf{p}_2$, koje iznose a i $\gamma/\rho n$, respektivno. Da bi dekompozicija bila odgovarajuća zahteva se takođe da matrica $-\mathbf{E}\mathbf{T}^{-1}\mathbf{B}+\mathbf{D}$ bude nesingularna, pa je potrebno odrediti njen rang (na primer *Gauss*-ovim eliminacionim metodom).

Ako se uspešno izvrši dekompozicija kontrolne matrice, bite provere parnosti moguće je izračunati na sledeći način

$$\mathbf{p}_1=(-\mathbf{E}\mathbf{T}^{-1}\mathbf{B}+\mathbf{D})^{-1}(-\mathbf{E}\mathbf{T}^{-1}\mathbf{A}+\mathbf{C})\mathbf{m}^T, \tag{3.18}$$

$$\mathbf{p}_2=-\mathbf{T}^{-1}(\mathbf{A}\mathbf{m}^T+\mathbf{B}\mathbf{p}_1^T). \tag{3.19}$$

Dekompoziciju kontrolne matrice moguće je izvršiti pre stvarnog slanja informacija (eng. *off line*), dok se kompleksnost kodera svodi na izračunavanje delova $\mathbf{p}_1$ i $\mathbf{p}_2$. U slučaju da raskorak ne postoji ($a=0$), potrebno je izračunati samo drugi deo kontrolnih bita. Dodatno, pokazuje se da koderi LDPC kodova za koje važi $a\leq O(\sqrt{n})$ imaju kompleksnost koja se povećava linearno sa dužinom koda. Procena bita provere parnosti obavlja se serijski, što potencijalno predstavlja nedostatak algoritma. Interesantno rešenje problema, bazirano na dijagonalizaciji blok matrica, predložio je skorije *Nozaki* u [70], gde je pokazano kako se kodovanje može izvršiti paralelno u dva nezavisna toka.

3.3 Dekodovanje LDPC kodova

Dekodovanje LDPC kodova najčešće se obavlja iterativno, pri čemu u toku jedne iteracije čvorovi *Tanner*-ovog grafa razmenjuju poruke. U zavisnosti od tipa poruka koje se razmenjuju iterativne dekodere moguće je, u najopštijem smislu, podeliti na dva načina i to na (i) dekodere koji razmenjuju tvrde (eng. *Hard-Decision*, HD) ili meke odluke (eng. *Soft-Decision*, SD) i (ii) na dekodere koji funkcionišu prema *bit-flipping* (BF) ili *message-passing* (MP) principu. Pritom su dve podele međusobno nezavisne. HD dekoderi razmenjuju jednobitne poruke, dok MD dekoderi razmenjuju višebitne poruke koje mogu predstavljati cele ili realne brojeve. BF

dekoderi (svejedno HD ili MD) u načelu računaju provere parnosti, pri čemu svaki čvor *Tanner*-ovog grafa svojim susedima šalju identične poruke, dok je u slučaju MP dekodera, dozvoljeno da poruke poslate susedima budu različite.

Najjednostavniji BF dekoder predložio je *Gallager* u svom originalnom članku [26] i njegov princip rada može se iskazati u sledećem.

- Invertovati vrednost svakog bita koji učestvuje u više od polovine nezadovoljenih provera parnosti.
- Ponavljati prethodni korak dok ne bude više nezadovoljenih provera parnosti, ili se ne dostigne zadati broj iteracija.

Prevedeno na opis *Tanner*-ovim grafom u toku jedne iteracije varijabilni čvorovi šalju svojim susedima vrednosti bita koji su im pridruženi, dok kontrolni čvorovi sabiraju po modulu 2 pristigle poruke i rezultat vraćaju svim svojim susedima. Iteracija se završava invertovanjem vrednosti bita koji su od svojih suseda primili više od pola jedinica. Kako se operacije invertovanja obavljaju istovremeno, to se predloženi algoritam još naziva i *paralelnim BF dekoderom*. Moguće je u toku jedne iteracije invertovati vrednost samo jednog bita koji ima najveći broj nezadovoljenih provera parnosti. Tada govorimo o *serijskom BF dekoderu* [33]. Proces BF dekodovanja moguće je usložniti tako da čvorovi razmenjuju i meke poruke, kako je to predloženo u [88], ili im je moguće pridružiti težinske faktore [89, 90]. Više informacija o BF dekoderima moguće je pronaći u [91]. U nastavku će MP dekoderi biti opisani na nešto formalniji način.

MD dekoder se takođe opisuje *Tanner*-ovim grafom, tako da čvorovi predstavljaju *procesorske jedinice* i često se označava petorkom $D = (\mathcal{M}, \mathcal{Y}, \Phi^{(v)}, \Psi^{(c)}, \bar{\Phi}^{(v)})$. Skup $\mathcal{M}$ definiše alfabet koji se koristi pri razmeni poruka između čvorova, dok $\mathcal{Y}$ predstavlja skup vrednosti koje mogu imati poruke pristigle iz kanala. Ako je kanal binarni tada je $\mathcal{Y} = \{\pm 1\}$. Preslikavanje $\Phi^{(v)} : \mathcal{M}^{\gamma} \rightarrow \mathcal{M}$ implementira se u varijabilnim čvorovima. Neka je $\mathbf{m}_v^{(\ell)} = \{m_1^{(\ell)}, m_2^{(\ell)}, \ldots, m_\gamma^{(\ell)}\}$ označen skup poruka koje stižu u varijabilni čvor v u toku iteracije ℓ. Tada se poruka koju varijabilni čvor v šalje preko grane e u toku iteracije ℓ, $\mu_e^{(\ell)}$, može izračunati kao

$$\mu_e^{(\ell)} = \Phi^{(v)}(\mathbf{m}_v^{(\ell-1)} \setminus m_e^{(\ell-1)}, y_v). \tag{3.20}$$

Slično, neka je $\mathbf{n}_c^{(\ell)} = \{n_1^{\ell}, n_2^{(\ell)}, \ldots, n_\rho^{(\ell)}\}$ vektor poruka pristiglih u kontrolni čvor c. Tada preslikavanje $\Phi^{(c)} : \mathcal{M}^{\rho-1} \rightarrow \mathcal{M}$ definiše vrednost poruke koju čvor c šalje preko grane e u

toku iteracije ℓ, $\nu_e^{(\ell)}$, na sledeći način

$$\nu_e^{(\ell)} = \Phi^{(c)}(\mathbf{n}_v^{(\ell)} \setminus n_e^{(\ell)}). \tag{3.21}$$

Primetiti da se prilikom računanja poruke koja se šalje preko neke grane e izostavlja poruka koja je u toku prethodne iteracije stigla preko iste grane e. U svakom varijabilnom čvoru treba implementirati γ $\Phi^{(v)}(\cdot)$ funkcija, dok se u kontrolnim čvorovima računa ρ $\Phi^{(c)}(\cdot)$ funkcija. Iterativni proces se završava u najviše L iteracija kada se na osnovu svih poruka pristiglih u varijabilni čvor v računa $\bar{\Phi}^{(v)} : \mathcal{M}^{\gamma+1} \to \mathcal{M}$, čiji izlaz daje procenu kodnog bita v, u oznaci $\hat{x}_v$, pa se definiše

$$\hat{x}_v = \bar{\Phi}^{(v)}(\mathbf{m}_v^{(L)}, y_v). \tag{3.22}$$

Konstruisanje različitih funkcija u čvorovima *Tanner*-ovog grafa obezbeđuje raznolikost MP dekodera. Tako se dekoderi dizajniraju na osnovu kompromisa kompleksnosti i željenog nivoa zaostale greške. Moguće je dekoder organizovati tako da se između čvorova prosleđuju verovatnoće (eng. *Belief Propagation*, BF), kada imamo $\mathcal{M} = \mathbb{R}$, dok se poruke koje razmenjuju čvorovi računaju na sledeći način

$$\mu_e^{(\ell)} = y_v + \sum_{e' \in \{\mathcal{E}(v) \setminus e\}} m_{e'}^{(\ell-1)}, \tag{3.23}$$

$$\nu_e^{(\ell)} = 2 \tanh^{-1} \Big(\prod_{e' \in \{\mathcal{E}(c) \setminus e\}} \tanh \Big(\frac{n_{e'}^{(\ell)}}{2} \Big) \Big), \tag{3.24}$$

$$\hat{x}_v = \begin{cases} +1, & \text{ako je } y_v + \sum_{e \in \mathcal{E}(v)} m_e^{(L)} > 0 \\ -1, & \text{inače.} \end{cases} \tag{3.25}$$

Osnovu dekodera čine operacije sabiranja i množenja, pa se često ovaj dekoder naziva SPA (eng. *Sum-Product Algorithm*) dekoderom. Treba istaći da u asimptotskom slučaju SPA dekoder predstavlja optimalan način dekodovanja, dok je za kodove konačne dužine samo suboptimalno rešenje. Uočiti da se operacije SPA dekodera obavljaju nad poljem realnih brojeva, što ovaj dekoder čini samo teorijski značajnim. Različite aproksimacije SPA dekodera predložene su u literaturi, od kojih je najznačajniji *min-sum* (MS) dekoder [92, 93]. Kompleksne operacije u kontrolnim čvorovima aproksimirane su na sledeći način

$$\nu_e^{(\ell)} \approx \Big(\prod_{e' \in \{\mathcal{E}(c) \setminus e\}} \operatorname{sgn}(n_{e'}) \Big) \min_{e' \in \mathcal{N}(c)} (|n'_e|), \tag{3.26}$$

pri čemu se operacije obavljaju u skupu $\mathcal{M} = \{-q, -(q-1), \ldots, -1, 0, 1, \ldots, q-1, q\}$, gde se vrednost q bira da zadovolji željeni nivo preciznosti. Kvantizovani iterativni dekoderi u opštem slučaju nazivaju se FAID (eng. *Finite Alphabet Iterative Decoder*). Interesantno je napomenuti da je moguće konstruisati FAID koji u nekim slučajevima prevazilazi performanse SPA dekodera. Više informacija o FAID moguće je pronaći u značajnim člancima [94, 95].

Kompleksnost dekodera moguće je dodatno uprostiti tako da čvorovi razmenjuju samo binarne poruke, tj. važi $\mathcal{M} = \{\pm 1\}$. Tada se poruke koje se razmenjuju između čvorova računaju kao

$$\mu_e^{(\ell)} = \begin{cases} -y_v, & \text{ako je } |e' \in \{\mathcal{E}(v) \setminus e\} : m_{e'}^{(\ell-1)}| \geq b^{(\ell)} \\ y_v, & \text{inače.} \end{cases} \tag{3.27}$$

$$\nu_e^{(\ell)} = \prod_{e' \in \{\mathcal{E}(c) \setminus e\}} n_{e'}^{(\ell)}. \tag{3.28}$$

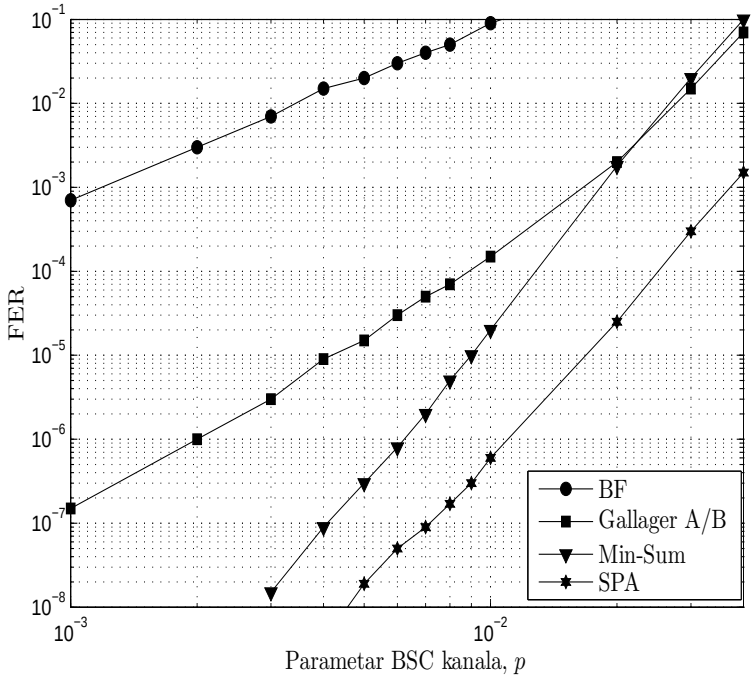
Navedeni dekoder naziva se *Gallager*-ovim dekoderom, pri čemu su posebno značajne dve verzije i to (i) kada je vrednost praga $b^{(\ell)} = \gamma - 1$, $\forall \ell \leq L$, što daje *Gallager A* dekoder i (ii) kada važi $b^{(\ell)} = \lceil \gamma/2 \rceil$, $\forall \ell \leq L$, što odgovara *Gallager B* dekoderu.

Performanse opisanih dekodera ilustrovane su na slici 3.3 na primeru koda QC(155,64) i BSC kanala za prenos. Performanse dekodera izražene su preko verovatnoće zaostale greške po kodnoj reči (eng. *Frame Error Rate*, FER). Uočljiva je superiornost SPA dekodera i veliki jaz između ovog teoretski najznačajnijeg dekodera i dekodera koji se mogu praktično koristiti. Razlog inferiornosti praktičnih kvantizovanih dekodera vezan je za pojavu *trapping set*-ova, koji su pomenuti ranije, ali će na ovom mestu biti formalno definisani.

Definicija 3.1. *Greška u varijabilnom čvoru v je ispravljiva ako postoji $L > 0$, tako da za svako $\ell \geq L$, važi $\hat{x}_v^{(\ell)} = x_v$, gde $\hat{x}_v^{(\ell)}$ predstavlja procenu bita x_v nakon ℓ iteracija dekodovanja.*

Neka je dat bipartitni graf $G' \subset G$, indukovan na osnovu skupa varijabilnih čvorova $\mathbf{T} \subset V$ tako da sadrži samo kontrolne čvorove koji su povezani na čvorove iz $\mathbf{T}$. Dalje je moguće formalno definisati pojam *trapping set*-a.

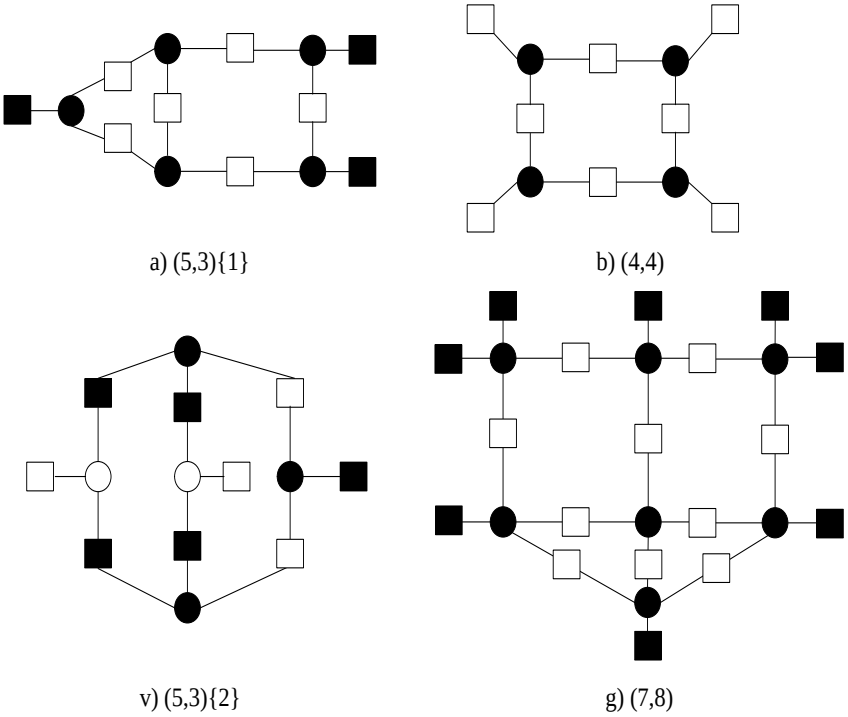
Definicija 3.2. *[96] Trapping set je skup varijabilnih čvorova Tanner-ovog grafa u kojima greške nisu ispravljive. Skup varijabilnih čvorova $\mathbf{T}$ naziva se trapping set (a, b), ako je $|\mathbf{T}| = a$ i postoji b kontrolnih čvorova iz G' koji su povezani sa neparnim brojem čvorova iz $\mathbf{T}$.*



Slika 3.3: Poređenje različitih iterativnih dekodera u BSC kanalu.

Definicija *trapping set*-a je generalna i uzima u obzir samo topologiju unutar *Tanner*-ovog grafa, iz čega sledi da struktura koja odgovara *trapping set*-u za jedan iterativni dekoder, ne mora nužno biti *trapping set* za drugi iterativni dekoder. Takođe, za neke kompleksne dekodere, kao SPA na primer, veoma je teško formirati profil karakterističnih *trapping set*-ova. Treba istaći da je *trapping set*-ove takođe teško identifikovati za slučajeve kada je $\gamma > 3$. Na slici 3.4 prikazane su neke karakteristične strukture koje su najčešći uzrok neuspešnog dekodovanja BF ili *Gallager*-ovog algoritma dekodovanja. Pri tome crni krugovi odgovaraju bitima koji su inicijalno pogrešno primljeni, dok beli krugovi označavaju isprtavno primljene bite. Beli kvadrati opisuju kontrolne čvorove koji imaju paran broj veza, dok crni kvadrati opisuju kontrolne čvorove sa neparnim brojem veza u *trapping set*-u. Treba napomenuti da broj čvorova u *trapping set*-u može biti veći od broja grešaka koje nije moguće ispraviti, što odgovara *kritičnom broju* (eng. *critical number*) *trapping set*-a. Tako, na primer, za slučaj *Gallager B* dekodera kritični broj $(5, 3)\{2\}$ *trapping set*-a iznosi tri.

Otkrivanje i klasifikacija *trapping set*-ova bila je predmet nekoliko značajnih radova dostupnih u literaturi [78, 97–104]. Tako je na primer Ivković [97] uočio da se performanse LDPC kodova u *error floor* regionu mogu uspešno proceniti, za zadati dekoder, otkrivanjem *trapping set*-a najmanje težine (tj. "dominantnih" *trapping set*-ova). Za zadatu verovatnoću greške u



Slika 3.4: Poređenje različitih iterativnih dekodera u BSC kanalu.

BSC kanalu p, u *error floor* regionu, verovatnoća greške po okviru $FER(p)$ se može iskazati kao

$$\log(FER(p)) = \log\Big(\sum_{i=t}^{n} c_i p^i (1-p)^{(n-i)}\Big) \approx \log(c_t) + t\log(p), \tag{3.29}$$

gde je c_i broj ulaznih sekvenci sa i pogrešno primljenih bita, koji dovodi do zaostale greške na izlazu iz dekodera. Vrednost c_t direktno je povezana sa brojem štetnih konfiguracija kod kojih je $|\mathbf{T}| = t$. Poznato je na primer da kod QC(155,64) sadrži 155 *trapping set*-ova označenih sa (5,3){2} i da *Gallager B* dekoder ne može ispraviti sve trostruke greške na ovom kodu [97]. Tada je na osnovu jednačine (3.29) moguće doći do numeričkih vrednosti $FER(0,001) \approx 1,55 \times 10^{-7}$, $FER(0,003) \approx 4 \times 10^{-6}$, $FER(0,005) \approx 2 \times 10^{-5}$ koje se slažu sa rezultatima dobijenim Monte Karlo simulacijom (slika 3.3).

Karimi i *Banihashemi* [98] su predložili algoritam kojim se efikasno procenjuje broj dominantnih *trapping set*-ova za zadatu kontrolnu matricu LDPC koda. S druge strane, grupa autora predvođena *Chilappagari*-jem [101–104] istraživala je međusobnu zavisnost *trapping set*-ova i pronašala uslove koje pod kojim je moguće garantovano ispravljanje grešaka, ako se koriste BF ili *Gallager*-ov algoritam dekodovanja.

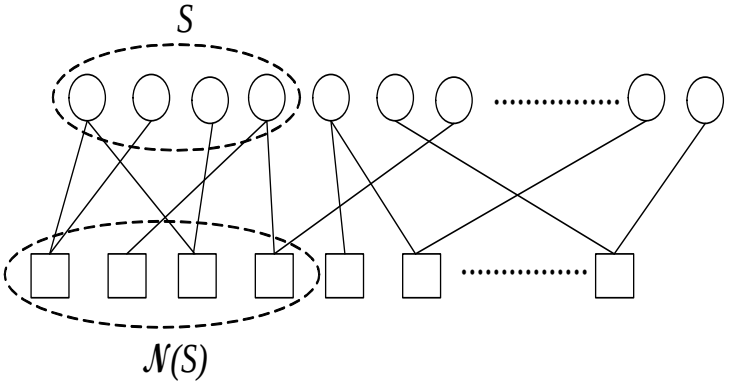
Pojam *trapping set*-ova veže se uvek za kanale koji unose greške, dok se kanali sa brisanjem opisuju *stopping set*-ovima. Za razliku od *trapping set*-ova, *stopping set*-ovi se mogu opisati

kombinatorički, što olakšava procenu performansi kodova. Za više informacija o dekodovanju na ovom tipu kanala čitaocu se preporučuje poznati članak [105].

3.4 Ekspander kodovi

Ekspander kodovi (eng. *expander codes*) predstavljaju teorijski jednu od najvažnijih klasa linearnih blok kodova. Razvili su ih *Sipser* i *Spielman* u značajnom radu [33] objavljenom 1996. godine, na osnovu originalnog *Gallager*-ovog [26] i *Margulis*-ovog [86] rada. Ekspander kodovi pripadaju klasi LDPC kodova i mogu se definisati nad ekspander grafovima čija je osobina da pojedini skupovi čvorova imaju neobično veliki broj suseda. U nastavku je data blago izmenjena definicija ekspander grafova, koja uzima u obzir ekspanziju samo varijabilnih čvorova *Tanner*-ovog grafa, što je ilustrovano na slici 3.5. Treba naglasiti da pojam ekspander grafova prevazilazi okvire LDPC kodova, ali će se u ovom odeljku fokus biti na osobinama karakterističnim za ovu primenu.

Definicija 3.3. Tanner-*ov graf G (γ, ρ)-regularnog LDPC koda je $(\gamma, \rho, \alpha, \delta)$ ekspander, ako za svaki skup varijabilnih čvorova S sa najviše αn čvorova, postoji najmanje $\delta|S|$ kontrolnih čvorova povezanih na čvorove iz S.*



Slika 3.5: Ilustracija uz definiciju ekspander grafova.

Definicija $(\gamma, \rho, \alpha, \delta)$ ekspandera podrazumeva da vrednosti α i δ ostaju konstantne kako se broj čvorova povećava. Graf se može nazvati "dobrim ekspanderom" ako je vrednost ekspanzije δ velika. Poznato je, na primer, da je slučajan (γ, ρ)-regularan graf dobar ekspander sa velikom verovatnoćom. Proces otkrivanja skupa čvorova proizvoljno izabranog grafa koji imaju određenu ekspanziju je NP-težak problem. Međutim, da li je neki graf dobar ekspander moguće

je proceniti na osnovu sopstvenih vrednosti matrice susedstva grafa. Ako je razlika između dve najveće sopstvene vrednosti velika, graf se može smatrati dobrim ekspanderom.

Izuzenu sposobnost ekspander grafova da ispravljaju greške uočio je *Zemor* [35] koji je formulisao teoremu datu u nastavku.

Teorema 3.1. *Za svako δ_0 koje zadovoljava $1 - 2H(\delta_0) > 0$, gde je $H(\cdot)$ entropijska funkcija [2], postoji ekspander kod kodnog količnika $1 - 2H(\delta_0)$ sa sposobnošću ispravljanja svake konfiguracije od $\alpha_0 n$ grešaka, $\alpha_0 < \delta_0^2/4$, dekoderom čija kompleksnost iznosi $O(n \log n)$.*

Iako je prethodni rezultat čisto asimptotski on otkriva da se korektivna sposobnost ekspander kodova linearno povećava sa dužinom koda – *kod ispravlja fiksnu frakciju grešaka* α_0. Drugim rečima teorema dokazuje da je moguće konstruisati kod koji ispravlja proizvoljno veliki broj grešaka, pri čemu kompleksnost iterativnog dekodera raste linearno sa dužinom koda (logaritamski član je posledica iterativnog postupka koji se obavlja u vremenu i ne utiče na hardversku kompleksnost). Navedena teorema predstavlja jedno od najvažnijih dostignuća teorije zaštitnog kodovanja, iako su *Barg* i *Zemor* kasnije uspeli da dokažu da ekspander kodovi dostižu *Shannon*-ov kapacitet na BSC kanalu [34]. Teorema 3.1 ima veći praktični značaj od *Shannon*-ove teoreme jer govori i o kompleksnosti dekodera, koji u asimptotskom slučaju ostvaruje savršenu pouzdanost. Iako je i sam *Gallager* pokazao da pod nekih uslovima kodovi iz $(\gamma > 2, \rho)$-regularnog ansambla u asimptotskom slučaju mogu postići proizvoljno malu verovatnoću zaostale greške, nije uspeo da dokaže da LDPC kodovi imaju sposobnost ispravljanja fiksne frakcije grešaka.

Treba istaći da u formulaciji teoreme nije precizirano da li sposobnost ispravljanja fiksne frakcije grešaka imaju praktično upotrebljivi iterativni dekoderi. *Sipser* i *Spielman* [33] su ispitali korektivnu sposobnost paralelenog BF dekodera, koja je predstavljena u teoremi datoj u nastavku.

Teorema 3.2. *Neka je dat $(\gamma, \rho, \alpha, (3/4 + \epsilon)\gamma)$, $\epsilon > 0$ ekspander. Paralelni BF dekoder može ispraviti svaku frakciju grešaka α_0, $\alpha_0 < \alpha(1 + 4\epsilon)/2$ nakon $\log_{1/(1-4\epsilon)}(\alpha_0 n)$ iteracija dekodovanja.*

Navedeni rezultat govori i o potrebi za određenom ekspanzijom grafa koja garantuje korektivnu sposobnost praktičnog dekodera. Treba navesti da su sličan rezultat *Sipser* i *Spielman* izveli i za serijski BF dekoder, kada se pokazuje da ispravljiva frakcija grešaka zadovoljava

[2] $H(x) := -x \log_2 x - (1 - x) log_2(1 - x)$

$\alpha_0 < \alpha/2$. Rezultate *Sipser*-a i *Spielman*-a generalizovali su *Burshtein* i *Miller* [36] analizirajući MP dekodere. Naime, pomenuti autori su uspeli da dokažu da za svaku ekspanziju $\delta > 3/4$ i *Gallager*-ov dekoder primenjen na $(\gamma > 5, \rho)$-regularne kodove ima sposobnost ispravljanja fiksne frakcije grešaka. S druge strane, *Feldman* [106] je pokazo da je koncept ekspander kodova primenljiv i na dekodere na bazi linearnog programiranja.

Eksplicitni metodi za konstrukciju ekspander kodova su retki. Poznato je, na primer, da se graf sa proizvoljnom ekspanzijom (ekspanzijom bliskom γ) može konstruisati pomoću slučajnih konstruktora i *zig-zag* proizvoda, kao što je to pokazao *Capalbo* [107]. Pri tome se garantuje postojanje grafa sa ekspanzijom δ ako je $\gamma = \text{poly}(\log(\gamma/\rho), 1/(1-\delta))$, što može dovesti do velike kompleksnosti koda. U literaturi su češći pokušaji da se ekspanzija grafova ograniči, ili da se samo dokaže postojanje ekspander grafa, bez pokušaja da se on i eksplicitno konstruiše. U nastavku će biti navedene najznačajnije teoreme koje opisuju postojanje ekspander grafova.

Teorema 3.3. *[106] Neka su $0 < r < 1$ i $0 < \delta < 1$ fiksne vrednosti, takve da je $(1-\delta)\gamma$ prirodan broj veći od jedan. Tada za svako n i m za koje važi $r = 1 - m/n$, postoji γ-levo-regularni Tanner-ov graf sa n varijabilnih čvorova i m kontrolnih čvorova koji je $(\gamma, \rho, \alpha, \delta)$ ekspander* [3] *gde je*

$$\alpha = \left[2e^{\delta\gamma+1}(\delta\gamma/(1-r))^{(1-\delta)\gamma}\right]^{-\frac{1}{(1-\delta)\gamma-1}}. \tag{3.30}$$

Teorema 3.4. *[104] Neka je G $(\gamma \geq 4)$-levo-regularni Tanner-ov graf girth-a g. Tada skup varijabilnih čvorova S iz G, $|S| < n_0(\gamma/2, g)$ ostvaruju ekspanziju bar $3\gamma/4$, pri čemu je*

$$\begin{aligned} n_0(\gamma/2, g_0) &= n_0(\gamma/2, 2j+1) = 1 + \frac{\gamma}{2}\sum_{i=0}^{j-1}\left(\frac{\gamma}{2}\right)^i, \ g_0 \ \textit{neparno}, \\ n_0(\gamma/2, g_0) &= n_0(\gamma/2, 2j) = 2\sum_{i=0}^{j-1}\left(\frac{\gamma}{2}\right)^i, \ g_0 \ \textit{parno}. \end{aligned} \tag{3.31}$$

Treba naglasiti da je Teorema 3.4 značajnija sa praktične strane, jer povezuje ekspanziju sa konstrukcionim parametrima γ i g, pri čemu kodove koji ispunjavaju uslove iz ove teoreme nije teško konstruisati. Teorema u nastavku ograničava broj suseda koje frakcija varijabilnih čvorova αn može da ima.

Teorema 3.5. *[33] Skup od αn varijabilnih čvorova Tanner-ovog grafa (γ, ρ)-regularnog koda, za svako $0 < \alpha < 1$ može imati najviše*

$$n\frac{\gamma}{\rho}(1-(1-\alpha)^\rho) + O(1), \ \textit{suseda.} \tag{3.32}$$

[3] definicija ekspandera u [106] je nešto drugačija i ne uzima u obzir težine vrsta kontrolne matrice ρ

Poslednja teorema daje maksimalne frakcije grešaka koje je moguće ispraviti nekim iterativnim dekoderom i predstavlja asimptotski pokazatelj kvaliteta dekodera.

3.5 Asimptotska *density evolution* analiza

Princip *density evolution* (DE) analize izložio je sam *Gallager*, dok su *Richardson* i *Urbanke* [5] ovu metodu sistematizovali i pomoću nje dokazali da LDPC kodovi dostižu *Shannon*-ov kapacitet. DE je moguće primeniti na MP dekoderima i to samo na dekoderima i kanalima koji ispunjavaju uslove simetričnosti, definisane u nastavku.

Definicija 3.4. Uslovi simetrije.

- *Kanal je simetričan ako važi*

$$p(y_v = q, x_v = 1) = p(y_v = -q, x_v = -1), \tag{3.33}$$

  za svako $q \in \mathcal{M}$ *i* $v \in V$.

- *Operacije u kontrolnim čvorovima su simetrične ako je*

$$\Phi^{(c)}(b_1 m_1, b_2 m_2, \ldots, b_{\rho-1} b_{m-1}) = \Phi^{(c)}(m_1, m_2, \ldots, m_{\rho-1}) \Big(\prod_{i=1}^{\rho-1} b_i \Big), \tag{3.34}$$

  za proizvoljno izabranu sekvencu $(b_1, b_2, \ldots, b_{\rho-1})$, $b_i \in \{\pm 1\}$.

- *Operacije u varijabilnim čvorovima su simetrične ako važi*

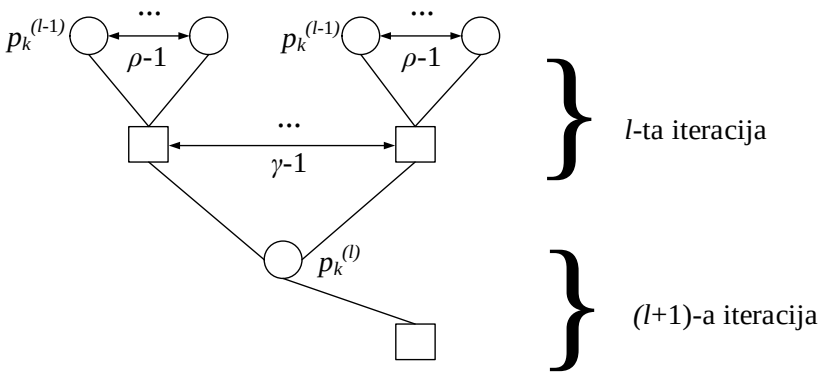
$$\Phi^{(v)}(-m_1, -m_2, \ldots, -m_{\gamma-1}, -y_v) = \Phi^{(v)}(m_1, m_2, \ldots, m_{\gamma-1}, y_v). \tag{3.35}$$

Lema data u nastavku govori o važnosti uslova simetrije u analizi iterativnih MP dekodera.

Lema 3.1. *Verovatnoća greške po bitu (okviru) LDPC koda dekodovanog MP dekoderom ne zavisi od kodne reči koja se prenosi, ako dekoder zadovoljava uslove simetrije date u Definiciji 3.4.*

Lako se pokazuje da BSC kanal, kao i svi MP dekoderi opisani u Odeljku 3.3 zadovoljavaju uslove simetrije, na osnovu čega sledi da je performanse dekodera moguće proceniti samo posmatrajnjem kodne reči sastavljene od jedinica. Neka je $p_k^{(\ell)}$ verovatnoća da poruke koje se šalju iz varijabilnih čvorova u toku iteracije ℓ imaju vrednost $k \in \mathcal{M}$. Tada verovatnoća da

je $k \neq 1$ dovodi do greške nakon dekodovanja. Cilj DE analize je upravo vezan za procenu verovatnoće zaostale greške po bitu nakon određenog broja iteracija ℓ. U tu svrhu potrebno je formirati stablo dubine 2ℓ, kako je to prikazano na slici 3.6, i propagirati verovatnoće pojave pogrešnih poruka od listova ka korenu stabla. Kako su poruke koje se propagiraju kroz stablo statistički nezavisne to će DE analiza dati tačnu procenu verovatnoće greške samo do iteracije $L = g/2$, gde g predstavlja *girth Tanner*-ovog grafa. Nakon L-te iteracije poruke koje se razmenjuju postaju korelisane, pa je problem određivanja verovatnoće greške NP-kompletan problem. Kako je često $g = 6$ ili $g = 8$, DE metod neće dati korektnu procenu verovatnoće greške praktičnog LDPC koda. Nepreciznost DE analize posebno je izražena u *error-floor* regionu [103]. S druge strane, može se smatrati da *Tanner*-ov graf beskonačno dugog koda ne sadrži cikluse, pa DE tehnika opisuje asimptotsko ponašanje (γ, ρ)-regulanog ansambla LDPC kodova (a ne posebnog koda). U nastavku će ukratko biti izložen DE postupak na primeru *Gallager A* dekodera. Sličnu analizu moguće je izvesti i za druge MP dekodere. Više informacija o DE analizi moguće je pronaći u člancima [5, 108].



Slika 3.6: Deo stabla korišćenog u DE analizi.

U *Gallager A* dekoderu razmenjuje se binarne poruke, $\mathcal{M} = \{\pm 1\}$, pa se verovatnoća pogrešne poruke u toku ℓ-te iteracije označava sa $p_{-1}^{(\ell)}$. Takođe važi $p_{-1}^{(0)} = \epsilon$, gde je ϵ verovatnoća greške u BSC kanalu. Tada se na osnovu zakona rada *Gallager A* dekodera može uspostaviti sledeća rekurzivna relacija

$$p_{-1}^{(\ell)} = p_{-1}^{(0)} - p_{-1}^{(0)}\left[\frac{1+\left(1-2p_{-1}^{(\ell-1)}\right)^{\rho-1}}{2}\right]^{\gamma-1} + (1-p_{-1}^{(0)})\left[\frac{1+\left(1-2p_{-1}^{(\ell-1)}\right)^{\rho-1}}{2}\right]^{\gamma-1}. \tag{3.36}$$

Indukcijom se može pokazati da $p_{-1}^{(\ell)}$ raste sa porastom $p_{-1}^{(0)}$. Tada je moguće pronaći maksimalnu vrednost $p_{-1}^{(0)} \in [0,1]$, označenu sa ϵ^*, za koju važi $\lim_{l\to\infty} p_{-1}^{(\ell)} = 0$. Vrednost ϵ^* predstavlja *prag šuma* koji garantuje proizvoljno malu verovatnoću greške u asimptotskom slučaju. Prag šuma se može numerički predstaviti za različite (γ, ρ)-regularne ansamble što je prikazano u tabeli datoj u nastavku.

Tabela 3.4: Vrednosti pragova šuma DE metode za *Gallager A* dekoder [5].

γ	ρ	Kodni količnik	ϵ^*
3	6	0,5	0,04
4	8	0,5	0,047
5	10	0,5	0,027
3	5	0,4	0,061
4	6	0,333	0,066
3	4	0,25	0,106

Posebno je značajno da su *Richardson* i *Urbanke* odredili vrednosti praga koje su bliske *Shannon*-ovom kapacitetu kanala sa brisanjem. Pored toga što DE analiza daje smernice kada je moguće postići proizvoljno malu verovatnoću greške, ona daje i minimalnu vrednost verovatnoće greške koju je moguće postići nekim iterativnim dekoderom nakon ℓ iteracija dekodovanja. Kako je SPA algoritam optimalan algoritam dekodovanja LDPC kodova kada ciklusi u grafu ne postoje, vrednosti $p_{-1}^{(\ell)}$ dobijene za slučaj SPA dekodera predstavljaju minimalnu verovatnoću greške nakon ℓ iteracija koju je moguće postići na konkretnom kodu konačne dužine, iz čega sledi da prisustvo ciklusa u *Tanner*-ovom grafu degradira performanse. Naravno verovatnoću greške koju SPA postiže na nekom konkretnom kodu moguće je unaprediti, jer su performanse SPA obično značajno lošije od optimalnih vrednosti.

Poglavlje 4

Bit-flipping dekodovanje nepouzdanim logičkim kolima

Bit-flipping (BF) tip dekodovanja predložio je *Gallager* u svom pionirskom članku [26] kao jednostavan i ilustrativan primer iterativnog dekodovanja u kome se vrednost primljenog bita invertuje ako bit učestvuje u većem broju nezadovoljenih provera parnosti. Tako razlikujemo (i) paralelni (PBF) i (ii) serijski (SBF) princip dekodovanja. U prvom slučaju u svakoj iteraciji paralelno se invertuju primljeni biti koji su povezani sa bar polovinom nezadovoljenih kontrolnih čvorova, dok se u drugom slučaju invertuju samo biti kod kojih je broj nezadovoljenih provera parnosti najveći.

Performanse *bit-flipping* dekodera nisu impresivne, ali je niska kompleksnost razlog praktičnog značaja ovakvih dekodera. Tokom godina, na bazi BF principa autori su dizajnirali čitav niz iterativnih dekodera opisanih *Tanner*-ovim grafovima koji svoje operacije baziraju na invertovanju kodnih bita [88–91, 109–112]. Tako je jednostavno pravilo invertovanja paralelnih/serijskih BF dekodera evoluiralo ka upotrebi "kriterijumske funkcije", koja pruža veću slobodu pri izboru bita koje treba invertovati, ali istovremeno usložnjava proces dekodovanja. Kriterijumska funkcija najčešće podrazumeva težinsko vrednovanje kodnih bita, pri čemu se težine biraju na osnovu mekih informacija iz kanala, kao u slučaju WBF (eng. *Weighted Bit-Flipping*) dekodera [89, 90, 111, 112]. Ovakave dekodere moguće je koristiti samo u kanalima čiji je kapacitet veći od kapaciteta binarnog simetričnog kanala, što predstavlja osnovni nedostatak pristupa. S druge strane, *Wadayama* [91] je dekodovanje posmatrao kao specijalni slučaj određivanja energetske beta funkcije čiji je argument kriterijumska funkcija – GDBF (eng. *Gradient-Descent BF*) dekoder. *Wadayama*-in optimizacioni postupak usmerava di-

namiku iterativnog dekodovanja ka lokalnom minimumu kriterijumske funkcije, što dovodi do uspešnog dekodovanja samo onda kada je lokalni minimum u isto vremene i globalni minimum. Da bi prevazišao manu originalnog algoritma i dinamiku procesa udaljio od lokalnog minimuma *Al Rasheed* je predložio da se odluke o invertovanju vrednosti bita ublaže, tako što će se biti invertovati sa određenom verovatnoćom – PGDBF (eng. *Probabilistic GDBF*) [109]. Kako je to primetio Ivaniš [110] ovaj stohastički proces moguće je unaprediti periodičnim reinicijalizacijama ulazne kodne sekvence, što je dovelo do algoritma u literaturi poznatog kao MUDRI (eng. *MUltiple-Decoding attempts and Random re-Initializations*). *Nguyen* i Vasić [88] su predložili da se poruke koje se razmenjuju između čvorova Tanerovog grafa ojačaju dodatnim bitom koji predstavlja procenu pouzdanosti poruke koja se prenosi – TBBF dekoderi (eng. *Two-Bit BF*). Dodatno, isti autori razvili su metod kolektivne zaštite koji podrazumeva paralelnu upotrebu više TBBF dekodera čije se korektivne sposobnosti dopunjuju.

Performanse iterativnih dekodera baziranih na invertovanju bita teško je analitički odrediti. Sa izuzetkom originalnih SBF i PBF dekodera, svi predstavljeni algoritmi su heuristike, čije se verovatnoće zaostale greške mogu proceniti jedino Monte Karlo simulacionim postupkom. S druge strane, u literaturi je poznato da SBF i PBF dekoderi, garantuju ispravljanje fiksnog broja grešaka. Tako su *Sipser* i *Spielman* u svom istaknutom članku [33] dokazali da ekspander kodovi dekodovani SBF ili PBF dekoderima mogu ispraviti *fiksnu frakciju grešaka*. Termin fiksna frakcija grešaka govori o postojanju nekog α, $0 < \alpha < 1$, takvog da dekoder može ispraviti svaku kombinaciju od αn grešaka, gde je n dužina kodne reči. Treba zapaziti da se broj grešaka koje je moguće israviti linearno povećava sa dužinom kodne reči, u graničnom slučaju potvrđujući zaključke II *Shannon*-ove teoreme. Slične, rezultate pružili su *Burshtein* i *Miller* [36] za *Gallager B* dekoder i *Feldman* [106] za dekoder na bazi linearne optimizacije (eng. *linear programming*). *Burshtein* je u svom kasnijem radu koristio optimizaciju entropijske funkcije da pokaže da svi LDPC kodovi iz $(\gamma \geq 4, \rho > \gamma)$-regularnog ansambla imaju sposobnost ispravljanja fiksne frakcije grešaka. Međutim, korektivne sposobnosti kodova sa težinom kolana $\gamma = 3$ znatno su skromnije i do sada nije poznato da li imaju sposobnost ispravljanja fiksne frakcije grešaka. S druge strane, poznato je da broj grešaka koje ispravljaju ovi kodovi raste linearno sa *girth*-om *Tanner*-ovog grafa, kako je to pokazao *Chilappagari* [101, 102]. Isti autor je takođe dokazao u [104] da za kodove sa $\gamma \geq 4$ broj grešaka koje PBF dekoder može ispraviti raste eksponencijalno sa *girth*-om *Tanner*-ovog grafa. Kako se *girth* logaritamski povećava sa dužinom kodne reči, autor nije uspeo dokazati ispravljanje fiksne frakcije grešaka, ali je pružio

donju granicu broja grešaka koje se garantovano mogu ispraviti ovakvim kodovima.

Iako je analiziran nezavisno od strane *Rudolph*-a [113] i *Massey*-ja [114] u kontekstu algebarske teorije kodovanja šezdesetih godina prošlog veka, dekoder na bazi majoritetnog odlučivanja (eng. *One-Step MAJority logic decoder*, OS-MAJ) predstavlja specijalni slučaj PBF dekodera u kome se dekodovanje zaustavlja nakon prve iteracije. Za razliku od iterativnih dekodera verovatnoća zaostale greške OS-MAJ dekodera može se proceniti analitički za kodove konačne dužine, kao što je to pokazao *Radhakrishnan* [115]. Značaj OS-MAJ dekodera se povećao sa reotkrivanjem LDPC kodova konstruisanih na osnovu konačnih geometrija [76], za koje OS-MAJ dekoder ima korektivne sposobnosti bliske ML tipu dekodovanja. Osim toga, niska kompleksnost i velika brzina dekodovanja omogućile su kombinovanje OS-MAJ dekodera sa drugim složenijim tipovima dekodovanja u kaskadnom lancu dekodera [75].

Zanimljivo je primetiti da osim analize OS-MAJ dekodera u prisustvu *von Neumann*-ovih grešaka, prezentovane u [116], u literaturi nema značajnijih rezultata vezanih za performanse BF dekodera napravljenih od nepouzdanih komponenata. U ovom poglavlju prezentovani su generalniji rezultati i performanse OS-MAJ dekodera analizirane su u prisustvu korelisanih grešaka koje je moguće modelovati *Markov*-ljevim lancem predstavljenim u Poglavlju 2. Izveden je matematički izraz u zatvorenoj formi za verovatnoću greške po bitu, koji važi za sve LDPC kodove koji ne sadrže cikluse dužine četiri. Korišćenje izvedene formule omogućava efikasniju procenu performansi različitih LDPC kodova, u poređenju sa računarski zahtevnom Monte Karlo simulacijom. Dodatno, istražene su sposobnosti iterativnih PBF dekodera, napravljenih od nepouzdanih komponenti, da ispravljaju fiksnu frakciju grešaka. Pokazano je da ekspander argumenti LDPC kodova predstavljaju dovoljan uslov za dokazivanje navedenog iskaza, u slučaju pojednostavljenog GOS modela grešaka. Takođe, analitički je određena i donja granica za broj grešaka koje se mogu ispraviti nepouzdanim PBF dekoderom. Rezultati predstavljeni u ovom poglavlju publikovani su u radovima [117, 118].

U Odeljku 4.1 predstavljena je arhitektura dekodera koji je analiziran u narednim odeljcima. Odeljak 4.2 posvećen je analizi OS-MAJ dekodera u prisustvu korelisanih grešaka opisanih genaralnim *Markov*-ljevim modelom. Specijalni slučaj GOS modela grešaka predstavljen je u Odeljku 4.3. Sposobnost PBF dekodera da ispravlja greške istražena je u Odeljku 4.4. Numerički rezultati koji ilustruju izvedene izraze dati su u Odeljku 4.5. Zaključci i potencijalni pravci daljeg istraživanja opisani su u Odeljku 4.6. Na kraju, u Dodacima su predstavljena duža matematička izvođenja.

4.1 Arhitektura nepouzdanog BF dekodera

Neka vektor $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ predstavlja kodnu reč LDPC koda, koja se nalazi na ulazu u binarni simetrični kanal. Izlaz iz kanala, označen sa $\mathbf{r} = (r_1, r_2, \ldots, r_n)$, gde je $\Pr\{r_k \neq x_k\} = p$, dekoduje se nepouzdanim BF dekoderom. Dodatno, broj gršaka koje je uneo kanal mogu se označiti preko *Hamming*-ovog rastojanja između binarnih vektora $\mathbf{x}$ i $\mathbf{r}$, u oznaci $d_H(\mathbf{x},\mathbf{r})$. Dekoder je podeljen na *procesorske jedinice* koje odgovaraju čvorovima u reprezentaciji koda *Tanner*-ovim grafom. Neka su $\overrightarrow{m}_i(e)$ i $\overleftarrow{m}_i(e)$ poruke koje se šalju preko grane e od varijabilnog čvora ka kontrolnom čvoru i od strane kontrolnog čvora ka varijabilnom čvoru, u toku i-te iteracije dekodovanja, respektivno. Slično, neka $\overrightarrow{m}_i(F)$ i $\overleftarrow{m}_i(F)$ predstavljaju skupove svih poruka koje se šalju od varijabilnog čvora, odnosno ka varijabilnom čvoru preko grana $F \subseteq E$, respektivno. U nastavku će ukratko biti izložena arhitektura BF dekodera, primenjenog na LDPC kod iz (γ, ρ)-regularnog ansambla.

- U toku iteracije $i = 0$ poruke koje varijabilni čvorovi šalju kontrolnim čvorovima inicijalizuju se porukama primljenim iz kanala, odnosno $\overrightarrow{m}_i(e) = r_v$, $\forall e \in \mathcal{N}(v)$. Za ostale iteracije i, $i > 0$, varijabilni čvor v donosi majoritetne odluke na skupu poruka poslatih od njegovih susednih kontrolnih čvorova

$$\Phi(\overleftarrow{m}_{i-1}(\mathcal{N}(v))) = \begin{cases} s, & \text{if } |\{e' \in \mathcal{N}(v) : \overleftarrow{m}_{i-1}(e') = s\}| \geq \lceil \gamma/2 \rceil, \\ r_v, & \text{inače}, \end{cases} \tag{4.1}$$

  gde je $s \in \{0, 1\}$, a $\lceil \gamma/2 \rceil$ označava najmanji prirodni broj veći od $\gamma/2$. Izlaz logičkog kola za većinsko odlučivanja (MAJ kolo), opisan funkcijom $\Psi(\cdot)$, prosleđuje se svim susednim kontrolnim čvorovima, $\overrightarrow{m}_i(e) = \Phi(\overleftarrow{m}_{i-1}(\mathcal{N}(v)))$, $\forall e \in \mathcal{N}(v)$.

- U toku iteracije i, $i \geq 0$, kontrolni čvor c vrši ρ ekskluzivnih-OR (XOR) operacija definisanih na sledeći način

$$\Psi(\overrightarrow{m}_i(\mathcal{N}(c) \setminus \{e\})) = \bigoplus_{e' \in \mathcal{N}(c) \setminus \{e'\}} \overrightarrow{m}_i(e'), \ \ \forall e \in \mathcal{N}(c). \tag{4.2}$$

  Rezultati XOR funkcija predstavljaju procene susednih varijabilnih čvorova i prosleđuju se mapiranjem $\overleftarrow{m}_i(e) = \Psi(\overrightarrow{m}_i(\mathcal{N}(c) \setminus \{e\}))$, $\forall e \in \mathcal{N}(c)$.

Ako se dekodovanje završi nakon i-te iteracije, rezultat $\Phi(\overleftarrow{m}_i(\mathcal{N}(v)))$ predstavlja procenu bita x_v. Treba primetiti da kada je dekoder napravljen od pouzdanih komponenti, iako kontrolni čvorovi procenjuju vrednost kodnih bita, a ne proveravaju parnost, predstavljeni dekoder je

ekvivalentan PBF dekoderu [33]. Međutim, mana standardne arhitekture u kojoj se računaju provere parnosti uočljiva je kada su prisutne greške u logičkim kolima. U tom slučaju otkaz jednog XOR logičkog kola utiče na ρ susednih varijabilnih čvorova, dok u predloženoj arhitekturi utiče samo na jedan susedni čvor. U ovom poglavlju se smatra da nepouzdanost dekodera potiče od nepouzdanog računanja operacije $\Psi(\cdot)$, jer su XOR kola korišćenja za njenu implementaciju inherentno podložna korelisanim greškama, opisanim u Poglavlju 2.

Kako se nakon svake iteracije kodni biti procenjuju na osnovu funkcije $\Phi(\cdot)$, to je i verovatnoća zaostale greške nakon dekodovanja ograničena sa donje strane verovatnoćom otkaza MAJ logičkog kola. Ako bi ova kola bila nepouzdana dekodovanje ne bi dalo željeni efekat. Zbog toga se zahteva da MAJ kola budu savršeno pouzdana ili da njihova nepouzdanost bude toliko niska da ne utiče značajno na verovatnoću zaostale greške. Zbog toga se u ovom poglavlju smatra da je izračunavanje funkcije $\Phi(\cdot)$ uvek pouzdano, a pouzdana MAJ logička kola se u literaturi često nazivaju "zlatnim logičkim kolima" [22, 25, 117]. Da bi se obezbedila njihova pouzdanost ona mogu biti izgrađena od većih tranzistora ili napajana većom snagom. Ako se dekodovanje stopira nakon prve iteracije, a kodni biti x_v procene na osnovu $\Phi(\overleftarrow{m}_0(\mathcal{N}(v)))$, BF dekoder se svodi na OS-MAJ dekoder, skorije analiziran u nekoliko relevantnih radova [116, 117].

4.2 Analiza OS-MAJ dekodera u prisustvu korelisanih otkaza logičkih kola

U ovom odeljku prezentovan je analitički metod za procenu performansi ansambla regularnih LDPC kodova, kod kojih je dužina najkraćeg ciklusa veća od četiri, dekodovanih OS-MAJ dekoderom, opisanim u prethodnom odeljku. U kodu čiji *Tanner*-ov graf ne sadrži cikluse jednake četiri ne postoje dva varijabilna čvora koja dele dve iste provere parnosti, što znači da binarno stablo proizvoljno izabranog varijabilnog čvora v sadrži uvek $(\rho - 1) \times \gamma$ drugih varijabilnih čvorova.

Kako su ulazi MAJ logičkog kola ujedno i izlazi iz XOR kola implementiranim u susednim kontrolnim čvorovima, to se u određenom bitskom intervalu k, varijabilnim čvorovima može pridružiti *vektor stanja* $\sigma^{(k)} = (\mathbf{s}_1^{(k)}, \mathbf{s}_2^{(k)}, \ldots, \mathbf{s}_\gamma^{(k)})$, čiji elementi predstavljaju stanja pojedinačnih susednih XOR kola. Takođe, na osnovu vektora $\sigma^{(k)}$, moguće je obrazovati i *vektor verovatnoća otkaza* $\varepsilon^{(k)} = (\varepsilon_1^{(k)}, \varepsilon_2^{(k)}, \ldots, \varepsilon_\gamma^{(k)})$, $\varepsilon_m^{(k)} = \Pr\{\xi^{(k)} = 1 | \mathbf{s}_m^{(k)}\}$, $1 \leq m \leq \gamma$. Vred-

nosti vektora verovatnoća greške moguće je proceniti merenjima ili simulacijama na tranzistorskom nivou željene tehnologije izrade. Tako je u analizi smatrano da su one poznate. U nastavku je prvo procenjena verovatnoća greške prilikom dekodovanja bita x_v za proizvoljno izabrani vektor stanja σ, odnosno vektor verovanoće greške ε, gde su indeksi vremenske odrednice izostavljeni radi jasnijeg zapisa.

Neka je sa $\mathbf{q}_l$ označen jedan leksikografski uređen vektor od u elemenata skupa $[l] = \{1, 2, \ldots, l\}$, i neka $\mathbf{q}_r$ sadrži preostale elemente $[l]$, proizvoljno raspoređene. Prostim spajanjem (eng. *juxtapositioning*) navedenih vektora $\mathbf{q}_l$ i $\mathbf{q}_r$ nastaje vektor $\mathbf{q}$. Vrste matrice $\mathbf{Q}^{u,l}$, dimenzija $\binom{l}{u} \times l$, sadrže sve moguće vektore $\mathbf{q}$. Na primer, ako je $l = 4$ i $u = 2$, vrste matrice $\mathbf{Q}^{2,4}$ su redom $(1, 2, 3, 4)$, $(1, 3, 2, 4)$, $(1, 4, 2, 3)$, $(2, 3, 1, 4)$, $(2, 4, 1, 3)$ and $(3, 4, 1, 2)$. Matrica $\mathbf{Q}^{u,l}$ se korsti za praćenje različitih kombinacija vektora verovatnoće greške, što je predstavljano lemom datom u nastavku.

Lema 4.1. *Verovanoća da je kodni bit x_v LDPC koda iz (γ, ρ)-regularnog ansambla pogrešno dekodovan nepozdanim OS-MAJ dekoderom, kome je pridružen vektor verovatnoće greške ε, data je sa*

$$P_v(p, \varepsilon) = \sum_{i=\lfloor \frac{\gamma+1}{2} \rfloor}^{\gamma} \sum_{j=1}^{\binom{\gamma}{i}} \prod_{m=1}^{i} P_{q_{j,m}} \prod_{m=i+1}^{\gamma} \left(1 - P_{q_{j,m}}\right) + \frac{(-1)^\gamma + 1}{2} p \sum_{j=1}^{\binom{\gamma}{\lfloor \frac{\gamma}{2} \rfloor}} \prod_{m=1}^{\lfloor \frac{\gamma}{2} \rfloor} P_{q_{j,m}} \prod_{m=\lfloor \frac{\gamma}{2} \rfloor + 1}^{\gamma} \left(1 - P_{q_{j,m}}\right), \tag{4.3}$$

gde je $P_{q_{j,m}} = \varepsilon_{q_{j,m}}(1 - A) + (1 - \varepsilon_{q_{j,m}})A$*,*

$$A = 0.5(1 - (1 - 2p)^{(\rho-1)}), \tag{4.4}$$

a $q_{t,m}$ *predstavlja element u preseku* t*-te vrste i* m*-te kolone matrice* $\mathbf{Q}^{i,\gamma}$*.*

Dokaz: Pogledati Dodatak 4.A. ■

Neka je sa $\{\mathbf{x}^{(k)}\}_{k \geq 0}$ označena sekvenca kodnih reči koje se šalju kroz kanal. Očigledno, moguća greška tokom dekodovanja $\mathbf{x}^{(k)}$ zavisi od $M - 1$ kodnih reči, prethodno poslatih kroz kanal, gde je sa M označen memorijski red *Markov*-ljevog modela grešaka, datog u prethodnom poglavlju. Dodatno, neka je $\mathbf{x}_{m,v} = \{\mathbf{x}_{m,v}^{(j)}\}_{j \in [k-(M-1),k]}$, $1 \leq m \leq \gamma$, $1 \leq v \leq n$, sekvenca kodnih bita, koja bi se pojavila na ulazu u m-to XOR kolo povezano sa varijabilnim čvorom v, ako kanal ne bi uneo ni jednu grešku u vremenskom intervalu $[k-(M-1), k]$. Sada

je moguće formulisati teoremu koja matematički opisuje performanse nepouzdanog OS-MAJ dekodera, čije se greške modeluju generalnim modelom stanja.

Teorema 4.1. *Prosečna verovatnoća greške po bitu koju postiže (γ, ρ)-regularni LDPC kod dekodovan nepouzdanim OS-MAJ dekoderom, pod uslovom da je poslata sekvenca $\{\mathbf{x}^{(j)}\}_{j\in[k-(M-1),k]}$ iznosi*

$$\bar{P}_e(\text{greška}|\mathbf{x}^{(k)},\ldots,\mathbf{x}^{(k-M+1)}) = \frac{1}{n}\sum_{v=1}^{n}\sum_{t=1}^{2^{(\rho-1)\gamma M}} P_v\left(p,\varepsilon^{(t)}\right) \times \prod_{m=1}^{\gamma} p^{d_H\left(\mathbf{s}_m^{(t)},\mathbf{x}_{m,v}\right)}(1-p)^{M(\rho-1)-d_H\left(\mathbf{s}_m^{(t)},\mathbf{x}_{m,v}\right)}. \quad (4.5)$$

Dokaz: Pogledati Dodatak 4.B. ■

Zapaziti da u opštem slučaju vektori verovatnoće greške zavise od prethodno poslatih kodnih reči, pa izraz (4.5) predstavlja uslovnu verovatnoću greške. Računarska kompleksnot izvedenog izraza raste eksponencijalno sa povećanjem levog i desnog stepena *Tanner*-ovog grafa, kao i memorijskog reda modela otkaza. Sa druge strane, različiti vektori verovatnoće otkaza, $\varepsilon^{(1)}, \varepsilon^{(2)}, \ldots, \varepsilon^{(t)}$, mogu dovesti do iste zaostale verovatnoće greške, $P_v\left(p,\varepsilon^{(1)}\right) = P_v\left(p,\varepsilon^{(2)}\right) = \ldots = P_v\left(p,\varepsilon^{(t)}\right)$, i u praksi broj sabiraka koje treba uzeti u obzir može biti značajno manji. Na primer, za značajni tip otkaza, modelovanih GOS modelom, potrebno je sabrati svega $\gamma + 1$ članova. Detaljna analiza ovog slučaja data je u narednom odeljku.

Za slučaj tranzijentnih grešaka, modelovanih *von Neumann*-ovim pristupom verovatnoća zaostale greške ne zavisi od vektora verovatnoće otkaza i imamo $P_v\left(p,\varepsilon^{(t)}\right) = P\left(p,\bar{\varepsilon}\right)$, $1 \leq t \leq 2^{(\rho-1)\gamma M}$, $1 \leq v \leq N$. Tako da se za ovakav specijalni slučaj izraz (4.5) svodi na formulu (4.3), što se dodatno može pojednostaviti prema metodu predloženom u [116]. Dodatno, ako sva XOR kola otkazuju sa istom verovatnoćom $\varepsilon_i = \bar{\varepsilon}$, $1 \leq i \leq \gamma$, svaka konfiguracija od i pogrešnih procena bita je podjednako verovatna i jednačina (4.3) se može pojednostaviti kao

$$P_v(p,\bar{\varepsilon}) = \sum_{i=\lfloor(\gamma+1)/2\rfloor}^{\gamma}\binom{\gamma}{i}P^i(1-P)^{\gamma-i} + \frac{(-1)^\gamma+1}{2}p\binom{\gamma}{\gamma/2}P^{\gamma/2}(1-P)^{\gamma/2}, \quad (4.6)$$

gde je $P = (1-A)\bar{\varepsilon} + A(1-\bar{\varepsilon})$.

4.3 Analiza OS-MAJ dekodera pri GOS modelu otkaza

Izlaz nekog XOR kola ostaje nepromenjen ako se ulazni vektori u dva uzastopna trenutka $k-1$ and k, $k>0$, razlikuju u neparnom broju pozicija. Tako, na primer, m-to XOR kolo, korišćeno za dekodovanje bita x_v, neće otkazati u trenutku k, ako poslati vektori $\mathbf{x}_{m,v}^{(k-1)}$ i $\mathbf{x}_{m,v}^{(k)}$ zadovoljavaju relaciju $d_H(\mathbf{x}_{m,v}^{(k-1)}, \mathbf{x}_{m,v}^{(k)}) = 0 \pmod 2$ i ne desi se ni jedna greška u kanalu. Slično, logičko kolo će imati pogrešan izlaz sa verovatnoćom $\bar{\varepsilon}$ ako su svi biti primljeni bez greške, a pritom važi $d_H(\mathbf{x}_{m,v}^{(k-1)}, \mathbf{x}_{m,v}^{(k)}) = 1 \pmod 2$. Naravno, parnost ulaznog vektora može biti promenjena pod uticajem grešaka nastalim u telekomunikacionom kanalu, kada kanal invertuje neparan broj bita u dva sukcesivna trenutka. Verovatnoća unije tih događaja se može odrediti kao

$$B = \sum_{j=0}^{\rho-2} \binom{2(\rho-1)}{2j+1} p^{2j+1}(1-p)^{2\rho-2j-3} = \frac{1}{2}\big(1-(1-2p)^{2(\rho-1)}\big). \tag{4.7}$$

Konačno zaključujemo da je izlaz logičkog kola pogrešan sa verovatnoćom $\bar{\varepsilon}B$ kada važi $d_H(\mathbf{x}_{m,v}^{(k-1)}, \mathbf{x}_{m,v}^{(k)}) = 0 \pmod 2$. Neka sva XOR kola, koja se koriste za dekodovanje bita x_v i zadovoljavaju navedenu osobinu formiraju skup $\mathcal{G}_v$. Slično, $\mathcal{H}_v$ sačinjavaju ostala XOR kola, tj. kola za koje važi $d_H(\mathbf{x}_{m,v}^{(k-1)}, \mathbf{x}_{m,v}^{(k)}) = 1 \pmod 2$. Očigledno je da je $\mathcal{G}_v \cup \mathcal{H}_v = [\gamma]$.

Prethodnu diskusiju vezanu za nepouzdana XOR kola moguće je proširiti i formalizovati kao što je to učinjeno u lemi koja sledi.

Lema 4.2. *Neka su $\mathbf{x}^{(k-1)}$ i $\mathbf{x}^{(k)}$ kodne reči koje se dekoduju u dva sukcesivna bitska intervala. Nepouzdani OS-MAJ dekoder ostvaruje najlošije performanse kada je kardinalni broj skupa $\mathcal{G}_v$, $|\mathcal{G}_v| = 0$, $1 \le v \le n$, dok najbolji slučaj odgovara dekodovanju uzastopnih kodnih reči kada je $|\mathcal{G}_v| = \gamma$, $1 \le v \le n$.*

Dokaz: Greška na izlazu iz XOR logičkog kola iz skupa $\mathcal{G}_v$ dešava se sa verovatnoćom $B\bar{\varepsilon}$, dok je verovatnoća pogrešnog izlaza pod uslovom da je XOR kolo element skupa $\mathcal{H}_v$ jednaka $(1-B)\bar{\varepsilon}$. Kako je $B < 0.5$ logičko kolo iz $\mathcal{H}_v$ češće ima pogrešan izlaz. Dokaz leme sledi iz činjenice da verovatnoća zaostale greške $P_v(p,\varepsilon)$ monotono raste sa povećanjem nepouzdanosti hardvera, tj. za svaka dva $\varepsilon^{(t_1)}$ i $\varepsilon^{(t_2)}$ koja zadovoljavaju relaciju $\varepsilon_m^{(t_1)} \le \varepsilon_m^{(t_2)}$, $1 \le m \le \gamma$, važi $P_v(p,\varepsilon^{(t_1)}) \le P_v(p,\varepsilon^{(t_2)})$. ■

Prethodna lema otkriva fundamentalnu osobinu OS-MAJ dekodera izgrađenog od nepouzdanih komponenti: *zavisnost od redosleda kodnih reči koje se dekoduju*. Moguće je primetiti

da dekodovanje dve iste kodne reči rezultuje najnižom verovatnoćom zaostale greške, dok će dve komplementarne kodne reči biti dekodovane sa najmanjom pouzdanošću.

OS-MAJ dekoder sačinjen od pouzdanih komponenti zadovoljava teoremu simetrije koja kaže da verovatnoća greške ne zavisi od kodne reči koja se šalje. Treba uočiti da sličan zaključak ne važi za neouzdani OS-MAJ dekoder, kada su hardverski otkazi posledica smanjenja napajanja logičkih kola.

Neka je kardinalni broj skupa $\mathcal{G}_v$, jednak $|\mathcal{G}_v| = t_v$. Verovatnoća zaostale greške, data izrazom (4.3), zavisi samo do broja nenultih elemenata vektora ε, ali ne i od njihovog redosleda. Tada je moguće pojednostaviti notaciju uvođenjem $\tilde{\varepsilon}^{(t)} = (\tilde{\varepsilon}_1^{(t)}, \tilde{\varepsilon}_2^{(t)}, \ldots, \tilde{\varepsilon}_\gamma^{(t)})$: vektora verovatnoće otkaza sa t ne-nultih elemenata. Sada je moguće pojednostaviti rezultat Teoreme 4.1, za sluačaj GOS modela otkaza logičkih kola.

Lema 4.3. *Verovatnoća da kodni bit x_v (γ,ρ)-regularnog LDPC koda bude pogrešno dekodovan nepouzdanim OS-MAJ dekoderom, za GOS model otkaza iznosi*

$$\bar{P}_v(t_v) = \sum_{t=0}^{\gamma} P_v\left(p, \tilde{\varepsilon}^{(t)}\right) \sum_{j=t_{min}}^{t_{max}} \binom{t_v}{j} \binom{\gamma - t_v}{t - j} B^{\gamma+2j-t_v-t}(1-B)^{t_v+t-2j}, \tag{4.8}$$

gde je $t_{min} = \max(t + t_v - \gamma, 0)$, *a* $t_{max} = \min(t_v, t)$.

Dokaz: Verovatnoća da j ne-nultih verovatnoća otkaza u $\tilde{\varepsilon}^{(t)}$ potiče iz skupa $\mathcal{G}_v$, a $t - j$ iz skupa $\mathcal{H}_v$ jednaka je $\binom{t_v}{j}\binom{\gamma-t_v}{t-j} B^{\gamma+2j-t_v-t}(1-B)^{t_v+t-2j}$. Suma svih mogućih kombinacija koje dovode do t ne-nultih verovatnoća otkaza predstavlja doprinos $P_v\left(p, \tilde{\varepsilon}^{(t)}\right)$ u konačnoj verovatnoći zaostale greške. Sumiranje svih $\gamma + 1$ mogućih vrednosti t dovodi do konačne verovatnoće greške. ■

Na osnovu Lema 4.2 i 4.3, moguće je izmeriti efekat grešaka kroz analitički određene granice performansi, kao što je to opisano lemom u nastavku.

Lema 4.4. *Verovatnoća greške po bitu (γ, ρ)-regularnog LDPC dekodovanog nepouzdanim OS-MAJ dekoderom, $\bar{P}_{e,GOS}$, ograničena je sa*

$$\sum_{t=0}^{\gamma} \binom{\gamma}{t} B^t(1-B)^{\gamma-t} P_v\left(p, \tilde{\varepsilon}^{(t)}\right) \leq \bar{P}_{e,GOS} \leq \sum_{t=0}^{\gamma} \binom{\gamma}{t} B^{\gamma-t}(1-B)^t P_v\left(p, \tilde{\varepsilon}^{(t)}\right). \tag{4.9}$$

Dokaz: Prema zaključcima Leme 4.2, donja granica se može dobiti postavljajući $t_v = \gamma$ u jednakost (4.8). Slično, gornju granicu je moguće izračunati zamenjujući $t_v = 0$. ■

Granice date u nejednakosti (4.9) određene su na osnovu uslova opisanih u Lemi 4.2 i one predstavljaju najmanje i najveće vrednosti verovatnoće greške po bitu. U opštem slučaju zavise

od nekoliko parametara γ, ρ, $\bar{\varepsilon}$ i p i mogu se razlikovati za više redova veličine. U Odeljku 4.5 granične vrednosti su numerički ilustrovane, za nekoliko praktično značajnih kodova.

4.4 Ispravljanje grešaka BF dekoderom sastavljenim od nepouzdanih logičkih kola

U ovom odeljku istražene su korektivne sposobnosti BF dekodera predstavljenog u Odeljku 4.1, čiji su kontrolni čvorovi izgrađeni od nepouzdanih komponenti. Pokazano je da se broj grešaka koje je moguće ispraviti povećava linearno sa dužinom kodne reči, kada *Tanner*-ov graf koda zadovoljava osobinu ekspanzije. Pritom, korišćenje su sledeće dve pretpostavke: (i) MAJ logička kola upotrebljena u dekoderu rade pouzdano, a otkazi XOR kola mogu se opisati GOS modelom otkaza i (ii) u toku prve iteracije ne događa se više od C_{XOR} otkaza logičkih kola. Razlozi za uvođene ovih pretpostavki navedeni su dalje u tekstu. U nastavku je formulisana teorema koja opisuje sposobnost nepouzdanog BF dekodera da ispravlja greške.

Teorema 4.2. *Neka je dat* $(\gamma, \rho, \alpha, (7/8+\epsilon)\gamma)$ *ekspander kod,* $1/8 \geq \epsilon > 0$*. BF dekoder sa nepouzdanim operacijama u kontrolnim čvorovima, primenjen na ovakav kod može da ispravi* $|V_1|$ *grešaka, gde je* $|V_1| < \Big(3(3+8\epsilon)\alpha n/32 - \sqrt{2}C_{XOR}\Big)$.

Dokaz: Neka je V_i skup pogrešnih kodnih bita (varijabilnih čvorova) na početku i-te iteracije. Tada, se skup pogrešnih kodnih bita na početku $(i+1)$-e iteracije (odnosno na kraju i-te iteracije), V_{i+1}, može podeliti na dva potskupa: (i) $(V_{i+1} \cap V_i)$, potskup kodnih bita koji su ostali pogrešni nakon i-te iteracije, i (ii) $(V_{i+1} \setminus V_i)$, potskup novokorumpiranih kodnih bita, tj. kodnih bita koji bili ispravni na kraju $(i-1)$-e iteracije, ali su invertovani u toku i-te iteracije. Neka je S_i skup svih bita koji su ispravljeni u toku $(i-1)$-e iteracije, a ostali su ispravni i nakon i-te iteracije. Kako su biti iz S_i promenili vrednost u toku $(i-1)$-te iteracije, na osnovu definicije GOS modela otkaza, sledi da biti iz ovog skupa mogu izazvati otkaze susednih XOR kola u toku i-te iteracije, što za posledicu ima pojavu pogrešnih procena bita povezanih na kontrolni čvor sa XOR kolima koja su otkazala. S druge strane, ako je vrednost bita ostala nepromenjena u toku $(i-1)$-e iteracije, bit neće izazvati otkaz susednih XOR kola.

Svaka pogrešna procena bita iz $V_{i+1} \setminus V_i$ posledica je povezanosti (preko zajedničkih kontrolnih čvorova) tog bita sa bitima iz skupa $V_i \cup S_i$. Navedeni zaključak sleduje iz zapažanja da svaki kontrolni čvor iz koga je poslata pogrešna procena bita, mora biti povezan sa još jednim bitom, koji je uzrok otkaza XOR kola. Tada, svaka pogrešna procena govori da je kontrolni

čvor povezan sa bar dva varijabilna čvora iz $V_i \cup S_i \cup V_{i+1}$. S druge strane, nema restrikcija za kontrolne čvorove koji prosleđuju ispravne procene bita – oni mogu imati za susede čvorove iz $V_{i+1} \setminus V_i$ ili čvorove izvan skupa $V_i \cup S_i \cup V_{i+1}$. Na osnovu jednačine (7.8) sledi da broj ispravnih procena svakog bita iz $V_{i+1} \setminus V_i$ ne može biti veći od $\gamma/2$, što znači da svaki čvor iz ovog skupa doprinosi sa najviše $\gamma/2$ različitih susednih kontrolnih čvorova u ukupnom broju suseda skupa $V_{i+1} \setminus V_i$. Neka broj suseda skupa $V_i \cup S_i$ iznosi $\delta\gamma|V_i \cup S_i|$, δ, $0 < \delta \leq 1$. Tada se dobija

$$|\mathcal{N}(V_i \cup S_i \cup V_{i+1})| \leq \delta\gamma|V_i \cup S_i| + \gamma/2|V_{i+1} \setminus V_i|. \tag{4.10}$$

Kodni biti ispravljeni u toku i-te iteracije (skup $V_i \setminus V_{i+1}$), kao i biti iz skupa S_i mogu biti povezani na sve različite kontrolne čvorove. Kako čvor iz $V_i \cap V_{i+1}$ deli bar polovinu suseda sa drugim čvorovima iz $V_i \cup S_i$, doprinosi sa najviše $3\gamma/4$ dodatnih čvorova u $\delta\gamma|V_i \cup S_i|$ pa se dobija

$$\begin{aligned}\delta\gamma|V_i \cup S_i| &\leq \gamma(|V_i| + |S_i| - |V_{i+1} \cap V_i|) + 3\gamma/4|V_{i+1} \cap V_i| \\ &= \gamma(|V_i| + |S_i|) - \gamma/4|V_{i+1} \cap V_i|.\end{aligned} \tag{4.11}$$

Ako se pretpostavi da važi

$$|V_i \cup V_{i+1} \cup S_i| < \alpha n \tag{4.12}$$

za svako $i > 0$, tada, na osnovu osobine ekspanzije sledi,

$$|\mathcal{N}(V_i \cup S_i \cup V_{i+1})| \geq (7/8 + \epsilon)\gamma(|V_i| + |S_i| + |V_{i+1} \setminus V_i|). \tag{4.13}$$

Kombinujući prethodni izraz sa jednakostima (4.10) i (4.11) dobija se

$$\begin{aligned}|V_i|(1 - 8\epsilon) &\geq (3 + 8\epsilon)|V_{i+1} \setminus V_i| \\ &\quad + 2|V_{i+1} \cap V_i| + (8\epsilon - 1)|S_i| \geq 2|V_{i+1}| - (1 - 8\epsilon)|S_i|.\end{aligned} \tag{4.14}$$

Kako su svi elementi iz S_i bili pogrešni pre $(i-1)$-te iteracije, zaključuje se da važi $|S_i| \leq |V_{i-1}|$, što, na osnovu prethodne nejednakosti, dovodi do

$$(1 - 8\epsilon)|V_i| \geq 2|V_{i+1}| - (1 - 8\epsilon)|V_{i-1}|. \tag{4.15}$$

Neka je $|V_2| \leq \beta|V_1|$, $\beta > 0$. Tada se vrednost $|V_i|$ može ograničiti kao u lemi datoj u nastavku.

Lema 4.5. *Broj pogrešnih bita na početku i-te iteracije dekodovanja, $i > 1$, $|V_i|$, ograničen je sa*

$$|V_i| \leq \frac{4\sqrt{1-8\epsilon} + (2\beta - 1 + 8\epsilon)(\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon})}{(1-8\epsilon)\sqrt{9-8\epsilon}} \Big(\frac{2\sqrt{1-8\epsilon}}{\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon}} \Big)^i |V_1|. \quad (4.16)$$

Dokaz: Pogledati Dodatak 4.C ■

Da bi se kompletirao ovaj deo dokaza Teoreme 4.2, potrebno je analizirati prvu iteraciju dekodovanja i ograničiti $|V_2|$. U lemi datoj u nastavku određena je gornja granica vrednosti $|V_2|$ izražena preko $|V_1|$ i C_{XOR}, broja otkaza XOR kola u toku prve iteracije.

Lema 4.6. *Broj pogrešnih kodnih bita na kraju prve iteracije, $|V_2|$, pod uslovom da važi $|V_1| < (3 + 8\epsilon)\alpha n/4$, ograničen je sa*

$$|V_2| \leq \frac{1-8\epsilon}{2}|V_1| + C_{XOR}. \quad (4.17)$$

Dokaz: Na osnovu analize prezentovane u [33], poznato je da dekoder izgrađen od pouzdanih komponenata u toku prve iteracije smanjuje broj pogrešnih bita na najviše $(1 - 4\delta)|V_1|$, za sve $1/4 \geq \delta > 0$. Prvi sabirak u izrazu (4.17) dobija se primećujući da je $\delta = 1/8 + \epsilon$. Drugi sabirak u izrazu (4.17) sledi iz činjenice da otkaz jednog XOR kola dovodi do greške u najviše jednom varijabilnom čvoru. ■

Kombinujući izraze (4.16) i (4.17) dobija se

$$|V_i| \leq \frac{4\sqrt{1-8\epsilon}|V_1| + 2(\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon})C_{XOR}}{(1-8\epsilon)\sqrt{9-8\epsilon}} \Big(\frac{2\sqrt{1-8\epsilon}}{\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon}} \Big)^i. \quad (4.18)$$

Iz prethodnog izraza sledi da se, za $\epsilon \in (0, 1/8]$, broj pogrešnih bita smanjuje tokom vremena, što nakon dovoljnog broja iteracija dovodi do korekcije svih inicijalno pogrešnih kodnih bita.

Primetiti da je opisana analiza izvedena pod pretpostavkom da važi $|V_i \cup V_{i+1} \cup S_i| < \alpha n$, za svako $i > 0$ (jednačina (4.12)). Da bi se verifikovala uvedena pretpostavka u nastavku je korišćena matematička indukcija.

Neka važi $|S_{i-1} \cup V_{i-1} \cup V_i| < \alpha n$. Tada je jednačina (4.18) zadovoljena za prvih $i - 1$ iteracija i možemo je koristiti da ograničimo $|V_{i-1}|$ i $|V_i|$. Pretpostavimo da važi suprotno, odnosno $|S_i \cup V_i \cup V_{i+1}| \geq \alpha n$. Tada, kako je poznato da važi $|S_i \cup V_i| < \alpha n$, mora postojati neko $D \subset V_{i+1} \setminus (V_i \cup S_i)$ za koje je $D \cup S_i \cup V_i = \alpha n$ i $|\mathcal{N}(D \cup S_i \cup V_i)| \geq (7/8 + \epsilon)\gamma\alpha n$. S druge strane, za neko δ, $7/8 + \epsilon \leq \delta \leq 1$, broj kontrolnih čvorova povezanih sa $D \cup S_i \cup V_i$ ograničen je sa

$$|\mathcal{N}(D \cup S_i \cup V_i)| \leq \delta\gamma(|S_i| + |V_i|) + \gamma/2(\alpha n - |S_i| - |V_i|). \quad (4.19)$$

Kombinujući prethodnu relaciju sa donjom granicom dobijenom na osnovu osobine ekspanzije, imamo

$$|S_i| + |V_i| \geq \frac{3+8\epsilon}{8\delta - 4}\alpha n \geq \frac{3+8\epsilon}{4}\alpha n. \tag{4.20}$$

S druge strane, kako je

$$|S_i| + |V_i| \leq |V_{i-1}| + |V_i|, \tag{4.21}$$

na osnovu jednačine (4.18) konačno se dobija

$$|V_1| \geq \Big[g_1(\epsilon)\frac{3+8\epsilon}{4}\alpha n - g_2(\epsilon)C_{XOR}\Big]\frac{1}{\sqrt{1-8\epsilon}}, \tag{4.22}$$

gde je

$$g_1(\epsilon) = \frac{(\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon})(1-8\epsilon)}{4(\sqrt{9-8\epsilon} + \sqrt{1-8\epsilon})}\Big(\frac{\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon}}{2\sqrt{1-8\epsilon}}\Big)^{i-1}, \tag{4.23}$$

i

$$g_2(\epsilon) = \frac{\sqrt{9-8\epsilon} - \sqrt{1-8\epsilon}}{2}. \tag{4.24}$$

Funkcija $g_1(\epsilon)$ monotono raste na intervalu $(0, 1/8]$, a minimalna vrednost funkcije na ovom intervalu zadovoljava $\min\limits_{0<\epsilon\leq 1/8}(g_1(\epsilon)) > 3/8$. Slično, maksimalna vrednost funkcije $g_2(x)$ na istom intervalu je $\max\limits_{0<\epsilon\leq 1/8}(g_2(\epsilon)) = \sqrt{2}$. Kako je $1/\sqrt{1-8\epsilon} > 1$ zaključuje se da je nejednakost (4.24) u kontradikciji sa inicijalnom pretpostavkom za $|V_1|$ datom u formulaciji teoreme. Tako mora važiti $|S_i \cup V_i \cup V_{i+1}| < \alpha n$ za svako $i > 2$. Kada je $i = 2$, formula (4.20) svodi se na

$$|V_1| \geq \Big[\frac{3+8\epsilon}{4}\alpha n - C_{XOR}\Big]\frac{2}{3-8\epsilon}, \tag{4.25}$$

što je takođe u kontradikciji sa inicijalnim zahtevom. Na kraju, uslov $|V_1 \cup V_2| < \alpha n$ sledi iz nejednakosti (4.17) i inicijalnog uslova za $|V_1|$. Ovim je dokaz teoreme završen. ■

U prethodnoj analizi pretpostavljeno je da su XOR kola nepouzdana, ali ne i MAJ logička kola. Ako bi se dozvolilo da MAJ kola budu nepouzdana, korekcija grešaka ne bi mogla biti garantovana. Navedeni zaključak se temelji na činjenici da odluke koje dovode do ispravljanja bita u toku jedne iteracije mogu biti poništene otkazima MAJ kola.

Treba primetiti da korektivna sposobnost dekodera zavisi ne samo od osobina ekspanzije *Tanner*-ovog grafa, već takođe i od broja otkaza XOR kola u toku prve iteracije (C_{XOR}). Ako

je broj otkaza u prvoj iteraciji preveliki, proces dekodovanja ne konvergira ka ispravnoj kodnoj reči. Na osnovu GOS modela otkaza sledi da C_{XOR} zavisi od stanja XOR kola u trenutku pre prve iteracije. Iako u opštem slučaju nije moguće kontrolisati broj otkaza XOR kola pre nego što dekodovanje počne, postoji praktičan način da se prevaziđe ovaj problem i omogući da C_{XOR} bude jednako nuli. Pre nego što dekoder počne dekodovanje nove kodne reči moguće je "naterati" tranzistore da dostignu stacionarno stanje, tako da prva iteracije bude pouzdana. To se praktično može postići usporavajući sistemski takt u prvoj iteraciji i dozvoljavajući stabilizaciju nivoa signala. Kako je frekvencija takta niža, nema vremenski zavisnih otkaza, što dovodi do C_{XOR}=0.

U nastavku su poređene korektivne sposobnosti nepouzdanog BF dekodera sa razultatima prezentovanim u [33], gde je posmatran pouzdani dekoder. Primećuje se da prisustvo otkaza smanjuje broj grešaka koje dekoder može ispraviti. Na primer, ako je ekspanzija *Tanner*-ovog grafa $(7/8+\epsilon)$, pouzdani dekoder ispravlja $9/16\alpha n$ grešaka, što je dvostuko veće od korektivne sposobnosti nepouzdanog dekodera. U graničnom slučaju kada je $\epsilon = 1/8$ nepouzdani dekoder ispravlja $3\alpha n/8$ grešaka, što je samo $3/8$ broja grešaka ispravljivih pouzdanim dekoderom.

Problem konsrukcije kodova koji zadovoljavaju uslov ekspanzije blizak γ, nazvani ekspanderi bez gubidaka (eng. *lossless expanders*), istraživao je *Capalbo* u [107], gde je pokazano da je željenu ekspanziju od $7/8 + \epsilon$ moguće dostići ako za levi stepen *Tanner*-ovog grafa važi $\gamma = \text{poly}(\log(\gamma/\rho), 8/(1-8\epsilon))$. Navedeno istraživanje dokazuje postojanje ekspander koda koji može ispraviti linearnu frakciju grešaka iako su operacije u kontrolnim čvorovima podložne korelisanim otkazima logičkih kola.

Sada će garantovano ispravljanje frakcije grešaka nepouzdanog BF dekodera biti prošireno na binarni simetrični kanal. Uslov potreban da obezbedi proizvoljno malu verovatnoću greške analiziran je u lemi datoj u nastavku.

Lema 4.7. *Neka verovatnoća greške binarnog simetričnog kanala zadovoljava relaciju* $p < 3(3+8\epsilon)\alpha/32$. *Tada za verovatnoću pogrešnog dekodovanja kodne reči* $(\gamma, \rho, \alpha, (7/8+\epsilon)\gamma)$ *ekspander koda dužine* n, $P^{(n)}$, *važi*

$$\lim_{n\to\infty} P^{(n)} = 0. \tag{4.26}$$

Dokaz: Dekoder će ispraviti sve frakcije grešaka jednake ili manje od p. U nastavku će biti pokazano da je verovatnoća pojave više od $(p+\Delta)n$, $3(3+8\epsilon)\alpha/32 > \Delta > 0$ grešaka u

binarnom simetričnom kanalu ograničena. Tako na osnovu *Chernoff*-ove granice [119] sleduje

$$Pr\{\textit{broj grešaka}/n > p + \Delta\} \leq e^{-D(p+\Delta||p)n}, \tag{4.27}$$

gde je $D(x||y) = x\log(x/y) + (1-x)\log((1-x)/(1-y))$ *Kullback-Leibler*-ova divergencija dva *Bernoulli*-jeva procesa parametara x i y, respektivno. Kako je $D(p+\Delta||p) > 2\Delta^2 n$, desna strana nejednakosti (4.27) eksponencijalno se smanjuje sa povećanjem dužine kodne reči, što asimptotski dovodi do proizvoljno male verovatnoće greške. ■

Drugačiji pogled na garantovano ispravljanje grešaka LDPC kodovima pružio je *Chilappagari* [104], koji je korektivnu sposobnost izrazio preko *girth*-a *Tanner*-ovog grafa. Teorema data u nastavku proširuje zaključke date u [104] za slučaj nepouzdanog dekodera.

Teorema 4.3. *Neka je dat LDPC kod sa γ-levo-regularnim Tanner-ovim grafom $\gamma \geq 8$ i girth-om $g = 2g_0$. Tada, BF dekoder napravljen od nepouzdanih komponenti može da ispravi $|V_1|$ grešaka ako je $|V_1| < 9n_0(\gamma/4, g_0)/32 - \sqrt{2}C_{XOR}$, a*

$$\begin{aligned} n_0(\gamma/4, g_0) &= n_0(\gamma/4, 2j+1) = 1 + \frac{\gamma}{4}\sum_{i=0}^{j-1}\left(\frac{\gamma}{4}\right)^i,\ g_0 \ \textit{neparno}, \\ n_0(\gamma/4, g_0) &= n_0(\gamma/4, 2j) = 2\sum_{i=0}^{j-1}\left(\frac{\gamma}{4}\right)^i,\ g_0 \ \textit{parno}. \end{aligned} \tag{4.28}$$

Dokaz: Da bi dokazali teoremu, koristimo sledeću lemu.

Lema 4.8. *Broj kontrolnih čvorova povezanih sa skupom varijabilnih čvorova V u γ-levo-regularnom Tanner-ovom grafu* girth-*a $g = 2g_0$ zadovoljava*

$$|\mathcal{N}(V)| \geq \gamma|V| - f(|V|, g_0), \tag{4.29}$$

gde $f(|V|, g_0)$ predstavlja maksimalni broj grana u proizvoljno izabranom grafu sa $|V|$ čvorova i girth-om g_0.

Dokaz: Pogledati [104]. ■

Na osnovu *Moore*-ove granice, poznato je da $n(\bar{d}, g_0)$ čvorova grafa srednjeg stepena $\bar{d} \geq 2$ i *girth*-a g_0 zadovoljava [120]

$$n(\bar{d}, g_0) \geq n_0(\bar{d}, g_0), \tag{4.30}$$

gde je $n_0(\bar{d}, g_0)$ definisano u jednačini (4.28). S druge strane, kako je $\gamma/4 \geq 2$ graf sa $|V| < n_0(\gamma/4, g_0)$ čvorova mora imati srednji stepen manji od $\gamma/4$. Tada, na osnovu definicije srednjeg stepena grafa imamo

$$f(|V|, g_0) < \gamma|V|/8. \tag{4.31}$$

Kombinujući prethodni izraz sa jednačinom (4.29) dobijamo $|\mathcal{N}(V)| > 7\gamma/8$. ■

Primetiti da je u [104] pokazano da $\gamma \geq 4$ predstavlja dovoljan uslov za garantovanu korekciju grešaka u kodu sa *Tanner*-ovim grafom *girth*-a g. Usled otkaza u XOR kolima veća ekspanzija je potrebna u poređenju sa pouzdanim dekoderom, ali svi ostali zaključci ostaju isti.

4.5 Numerički rezultati

4.5.1 Analiza verovatnoće greške OS-MAJ dekodera

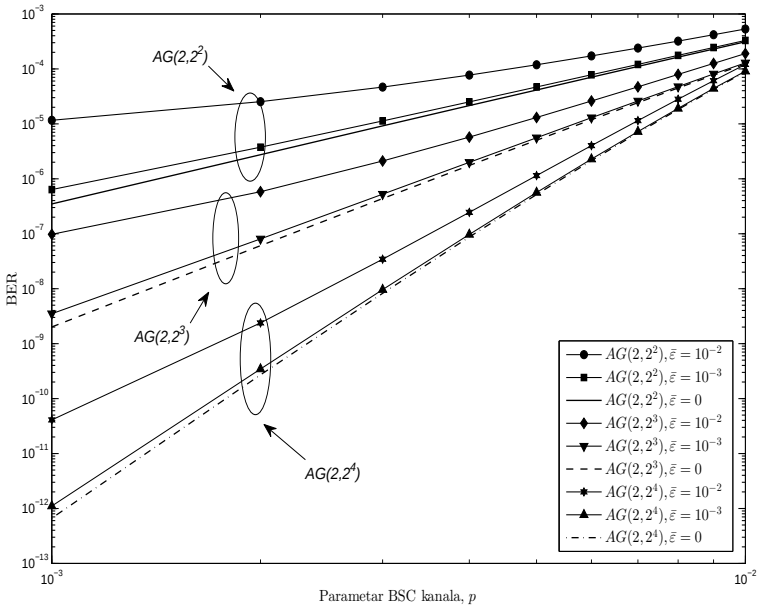
Kodovi dizajnirani na osnovu konačnih geometrija su značajni u kontekstu OS-MAJ dekodera [75]. Poznato je da se kodom izvedenim na osnovu konačnih geometrija može ispraviti $\lfloor \gamma/2 \rfloor$ grešaka pouzdanim OS-MAJ dekoderom. U ovom odeljku istražene su verovatnoće zaostale greške 2-dimenzionih afinih kodova i kodova na osnovu projektivne geometrije, formiranih na osnovu *Galois*-ovog polja GF(2^s), označenih sa AG($2, 2^s$) i PG($2, 2^s$), $s > 0$, respektivno. *Tanner*-ov graf afinih kodova AG($2, 2^s$) ima desni-stepen $\rho = 2^s + 1$, levi-stepen $\gamma = 2^s$ i minimalno kodno rastojanje $d_{min} = 2^s + 1$. Kodovi PG($2, 2^s$) opisani su sa $\rho = \gamma = 2^s + 1$, a imaju minimalno rastojanje $d_{min} = 2^s + 2$.

Prosečne verovatnoće greške po bitu (eng. *Bit Error Rate*, BER) za nekoliko PG i AG kodova, u prisustvu otkaza u XOR kolima opisanih GOS modelom, prikazane su na slici 4.1. Gornje granične vrednosti računate su pomoću izraza (4.9), za dve vrednosti verovatnoće otkaza $\bar{\varepsilon} = 10^{-3}, 10^{-2}$ i poređene sa pouzdanim dekoderom za koji je $\bar{\varepsilon} = 0$. Treba primetiti da donja granica verovatnoće greške odgovara slučaju veoma retkih otkaza i može se proceniti kao

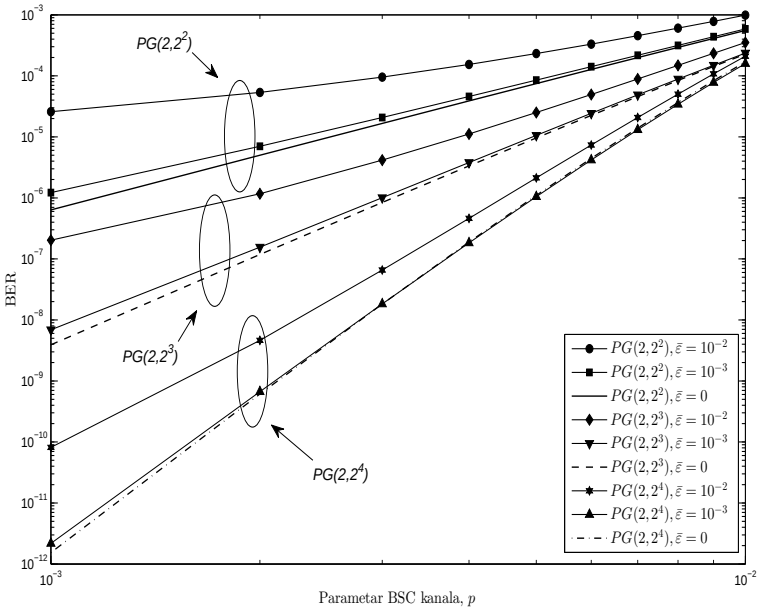
$$\sum_{t=0}^{\gamma} \binom{\gamma}{t} B^t (1-B)^{\gamma-t} P_v\left(p, \bar{\varepsilon}^{(t)}\right) \approx P_v\left(p, (0, \ldots, 0)\right). \tag{4.32}$$

Zbog toga su donje granice izostavljene sa slike 4.1.

Potrebno je uočiti da česti otkazi logičkih kola dovode do značajne degradacije performansi. Degradacija je posebno izražena u regionu sa niskom verovatnoćom greške u kanalu.



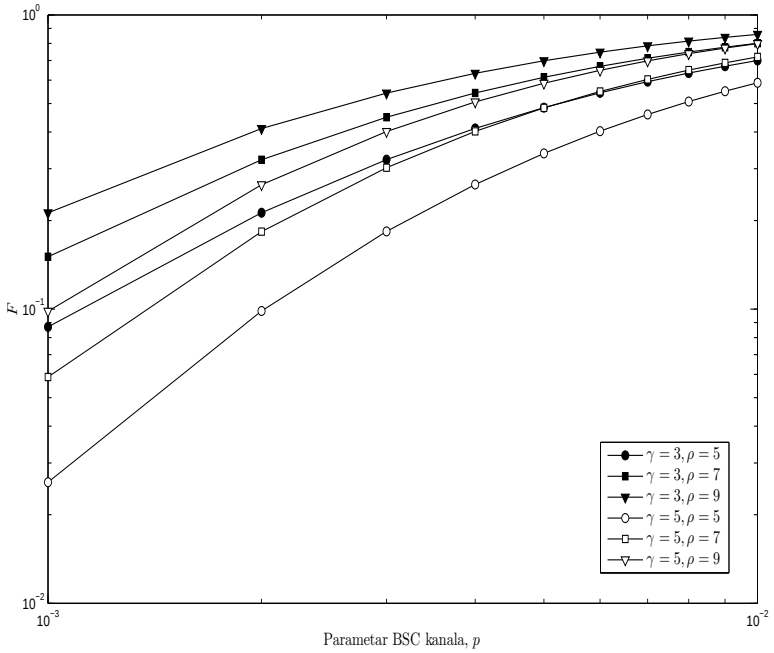
(a) AG kodovi



(b) PG kodovi

Slika 4.1: Analitički procenjene verovatnoće greške po bitu (BER).

Na primer, ako je $p = 10^{-3}$, ekstremno visoka nepouzdanost hardvera (izražena verovatnoćom otkaza $\bar{\varepsilon} = 10^{-2}$) povećava BER za čitav red veličine, za sve posmatrane kodove. S druge strane, verovatnoća otkaza jednaka $\bar{\varepsilon} = 10^{-3}$, dovodi do značajno manjeg gubitka pouzanosti. Gubitak se smanjuje sa povećanjem dužine koda (parametra s), što rezultuje zanemarljivo



Slika 4.2: Faktor korelacije otkaza za različite (γ, ρ)-regularne kodove ($\varepsilon = 10^{-2}$).

malom degradacijom BER vrednosti za kodove konstruisane za $s = 4$, odnosno AG$(2, 2^4)$ i PG$(2, 2^4)$. Kako je $\bar{\varepsilon} = 10^{-3}$ veoma velika verovatnoća otkaza, otpornost OS-MAJ dekodera na nepouzdanost hardvera je visoka. Za vrednosti $\bar{\varepsilon}$ ($\bar{\varepsilon} < 10^{-3}$) degradacija performansi je zanemarljiva za sve posmatrane kodove.

Kao zgodnu meru varijacija performansi moguće je definisati *faktor korelacije otkaza*, F, kao odnos graničnih vrednosti verovatnoće greške, datih u Lemi 4.4, na sledeći način

$$F = \frac{\sum_{t=0}^{\gamma} \binom{\gamma}{t} B^t (1-B)^{\gamma-t} P_v\left(p, \tilde{\varepsilon}^{(t)}\right)}{\sum_{t=0}^{\gamma} \binom{\gamma}{t} B^{\gamma-t} (1-B)^t P_v\left(p, \tilde{\varepsilon}^{(t)}\right)}. \tag{4.33}$$

Vrednosti F za različite (γ, ρ)-regularne LDPC kodove prikazane su na slici 4.2. Uočljivo je da je degradacija veća za veće vrednosti γ. Tako na primer, kada je $p = 10^{-3}$, za kodove opisane parametrima $\gamma = \rho = 5$, gornja granica performansi je više od sedamdeset puta veća od donje granice. Kako se korektivna sposobnost koda povećava sa γ, interesantno primetiti da su bolji kodovi osetljiviji na negativne efekte hardverske nepouzdanosti, za iste vrednosti parametra ρ. Dodatno, gubitak performansi se smanjuje sa povećanjem stepena kontrolnih čvorova.

4.5.2 Garantovano ispravljenje grešaka

Na osnovu Teoreme 4.2 sledi da broj grešaka koji je moguće ispraviti zavisi od osobine ekspanzije *Tanner*-ovog grafa izražene preko parametara α and ϵ, kao i otkaza nasleđenih pre početka dekodovanja, C_{XOR}. U ovom odeljku procenjene su granične vrednosti za frakciju grešaka kanala koje je moguće ispraviti BF dekoderom, $\alpha_{total} = 3(3+8\epsilon)\alpha/32 - \sqrt{2}C_{XOR}/n$. Lema data u nastavku numerički opisuje ovu granicu.

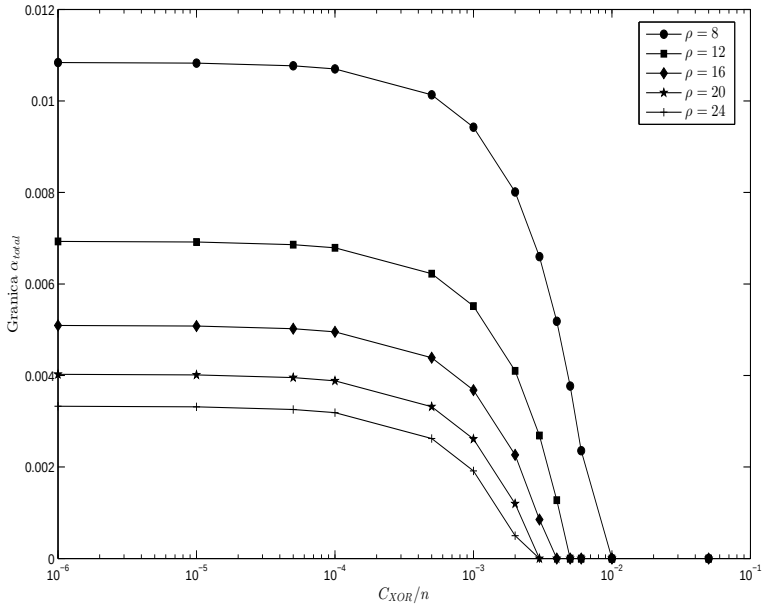
Lema 4.9. *Neka su α^* i ϵ^* takvi da važi $\alpha_{total}(\alpha^*, \epsilon^*) \geq \alpha_{total}(\alpha, \epsilon)$, $0 < \alpha < 1$, $0 < \epsilon \leq 1/8$. Tada, u graničnom slučaju kada dužina koda teži beskonačnosti važi*

$$\epsilon^* = (1-(1-\alpha^*)^\rho)/(\alpha^*\rho) - 7/8. \tag{4.34}$$

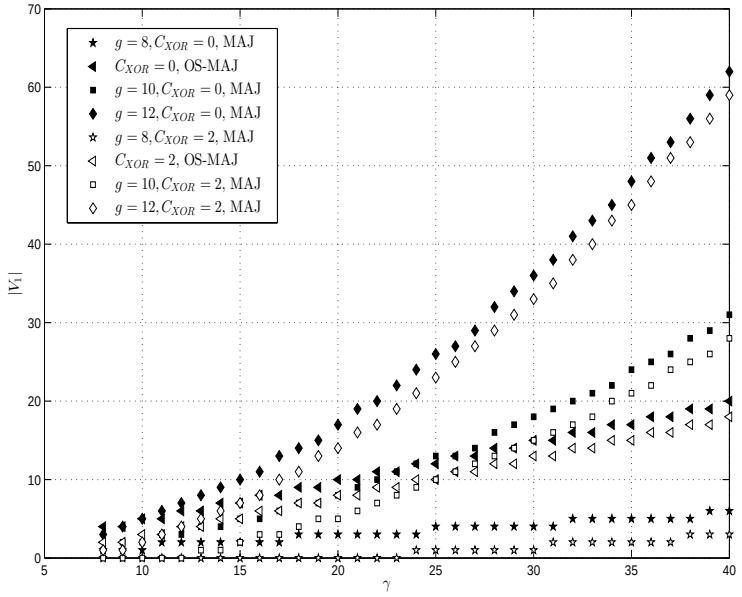
Dokaz: Prethodna relacija sledi iz [33, Theorem 25], gde je pokazano da αn varijabilnih čvorova ima maksimalno $n\gamma(1-(1-\alpha)^\rho)/\rho + O(1)$ suseda i činjenice da se pretraga obavlja za grafove koji ostvaruju ekspanziju od makar $(7/8+\epsilon)$. ■

Na slici 4.3(a) numerički je izražena vrednost $\alpha_{total}(\alpha^*, \epsilon^*)$ u zavisnosti od odnosa C_{XOR}/n, za različite ρ-desno-regularne *Tanner*-ove grafove. Razmatrani su slučajevi kada je $\rho \geq 8$. Moguće je primetiti da, kada je $\rho = 8$, a uticaj nasleđenih otkaza se može zanemariti, potencijalno se ispravlja 1% pogrešnih bita. Dodatno, korektivna sposobnost dekodera se smanjuje sa povećanjem parametra ρ. Kada je procenat nasleđenih otkaza uporediv sa korekcionom sposobnošću koda dostiže se *prag* nakon čega se frakcija ispravljivih grešaka naglo smanjuje. Uočljivo je da je prag nezavistan od ρ. Za dovolno veliko C_{XOR}/n performanse dekodera se degradiraju sve do trenutka kada nije moguće garantovano ispravljanje grešaka. Ovo se dešava na primer za $\rho = 8$, kada je $C_{XOR}/n \geq 1\%$.

Drugačiji pogled na korektivne sposobnosti nepouzdanog dekodera date su na slici 4.3(b). Ovde je ispitano kako *girth* utiče na performanse γ-levo-regularnog *Tanner*-ovog grafa. Dodatno, upoređeni su rezultati dati u Teoremi 4.3 sa korektivnom sposobnosti nepouzdanog OS-MAJ dekodera, izražene preko $\lfloor \gamma/2 \rfloor - C_{XOR}$. Uočljivo je da granica data Teoremom 4.3, za male vrednosti *girth*-a ($g \leq 8$), nije upotrebljiva. Ona u stvari daje lošije rezultate od poznate korektivne sposobnosti OS-MAJ dekodera. S druge strane, za veće vrednosti *girth*-a *Tanner*-ovog grafa, rezultati dati u Teoremi 4.3 su značajni. Na primer, kada je $g = 12$, $C_{XOR} = 0$ i $\gamma = 12$, moguće je garantovati ispravljanja svih sedmostrukih grešaka, što nije moguće korišćenjem OS-MAJ dekodera.



(a) Maksimalna frakcija ispravljivih grešaka



(b) broj ispravljivih grešaka

Slika 4.3: Korektivna sposobnost LDPC kodova.

4.6 Zaključak

Na osnovu uvedenog *Markov*-ljevog modela otkaza logičkih kola razvijen je matematički aparat koji omogućava procenu performansi OS-MAJ dekodera izgrađenih od nepouzdanih kompo-

nenti. Predloženi metod moguće je koristiti za procenu verovatnoće zaostale greške za sve kodove čiji *Tanner*-ovi grafovi ne sadrže cikluse dužine četiri. Uočeno je da su verovatnoće greške zavisne od redosleda kodnih reči koje se dekoduju, a granične vrednosti numerički su predstavljene za slučaj pojednostavljenog GOS modela otkaza logičkih kola.

Dodatno, na osnovu osobine ekspanzije *Tanner*-ovog grafa, uspostavljeni su uslovi potrebni da se korektivna sposobnost BF dekodera povećava linearno sa dužinm kodne reči. Iako je pokazano da navedena osobinu važi samo za kodove sa velikim levim- i desnim-stepenima grafa, rezultati predstavljeni u ovom poglavlju predstavljaju prve poznate rezultate o korektivnoj sposobnosti iterativnih dekodera napravljenih od nepouzdanih komponenti. Argumenti ekspanzije su se pokazali adekvatnim i za analizu *message-passing* iterativnih dekodera [36]. Bilo bi izazovno nastaviti rad *Burshtein*-a i *Miller*-a o neregularnim LDPC kodovima i omogućiti procenu korektivne sposobnosti *message-passing* dekodera izgrađenih od nepouzdanih komponenti.

Dodatak 4.A (dokaz Leme 4.1)

Verovatnoća da se bit r_v pogrešno dekoduje, pod pretpostavkom fiksnog vektora ϵ, moguće je izraziti kao

$$P_v(\alpha, \epsilon) = \sum_{N_e=0}^{n} \text{Pr}\{\hat{r}_v \neq x_v | N_e \text{ grešaka}\}\text{Pr}\{N_e \text{ grešaka}\}, \tag{4.35}$$

gde je $\text{Pr}\{\hat{r}_v \neq x_v | N_e \text{ grešaka}\}$ uslovna verovatnoća da je bit pogrešno dekodovan ako je kanal uneo greške na N_e kodnih bita, dok je verovatnoća tog događaja označena sa $\text{Pr}\{N_e \text{ grešaka}\}$. Zarad jednostavnijeg prikaza, a bez gubljenja generalnosti moguće je smatrati da je $x_v = 0$.

Na osnovu jednačine (7.8) sledi da se vrednost primljenog bita r_v koristi u procesu dekodovanja samo kada je γ parno. Tada se prethodni izraz može podeliti na dva dela na sledeći način

$$\begin{aligned} P_v(\alpha, \epsilon) = \sum_{N_e=0}^{n-1} [&\text{Pr}\{\hat{r}_v = 1 | N_e \text{ grešaka}, r_v = 0\}\text{Pr}\{r_v = 1 | N_e \text{ grešaka}\} \\ &+ \text{Pr}\{\hat{r}_v = 1 | N_e - 1 \text{ grešaka}, r_v = 1\}\text{Pr}\{r_v = 1 | N_e \text{ grešaka}\}]\text{Pr}\{N_e \text{ grešaka}\}. \end{aligned} \tag{4.36}$$

Primećujući da se N_e grešaka u kanalu (izuzimajući poziciju bita x_v) događaju sa verovatnoćom

$\binom{n-1}{N_e}\alpha^{N_e}(1-\alpha)^{n-1-N_e}$, prethodni izraz može dalje modifikovati

$$P_v(\alpha,\epsilon) = \sum_{N_e=0}^{n-1}\binom{n-1}{N_e}\alpha^{N_e}(1-\alpha)^{n-1-N_e} \left[(1-\alpha)\sum_{i=0}^{N_e}\text{Pr}\{\hat{r}_v=1|N_e^v=i,r_v=0\}\text{Pr}\{N_e^v=i|N_e \text{ grešaka}, r_v=0\} + \alpha\sum_{i=0}^{N_e}(\hat{r}_v=1|N_e^v=i,r_v=1)\,\text{Pr}\{N_e^v=i|N_e \text{ grešaka}, r_v=1\}\right], \tag{4.37}$$

gde N_e^v predstavlja broj pogrešno primljenih bita u binarnom stablu varijabilnog čvora v. Verovatnoća da je i čvorova u binarnom stablu pogrešno primljena nezavisna je od bita r_v, pa se tako dobija $\text{Pr}\{N_e^v=i|N_e \text{ grešaka}, r_v=0\} = \text{Pr}\{N_e^v=i|N_e \text{ grešaka}, r_v=1\} = A(N_e,i)$, što se može izraziti na sledeći način

$$A(N_e,i) = \begin{cases} \dfrac{\binom{n-\gamma(\rho-1)-1}{N_e-i}}{\binom{n-1}{N_e}} & \text{ako je } \le(\rho-1)\wedge N_e-1\le n-1-\gamma(\rho-1), \\ 0 & \text{inače.} \end{cases} \tag{4.38}$$

Ako se obe sume date u (4.37) razviju i sakupe koeficijenti uz $\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=0\}$ i $\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=1\}$ dobija se

$$P_v(\alpha,\epsilon) = \sum_{i=0}^{\gamma(\rho-1)}\alpha^i(1-\alpha)^{\gamma(\rho-1)-i}\Big[(1-\alpha)\,\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=0\} + \alpha\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=1\}\Big], \tag{4.39}$$

gde $\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=1\}$ i $\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=0\}$ predstavljaju verovatnoće pogrešnog dekodovanja pod uslovom da je i čvorova binarnog stabla pogrešno, a vrednost bita r_v jednaka jedinici i nuli, respektivno. Ove verovatnoće obrazuju sledeću relaciju

$$\text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=1\} = \sum_{t=\lfloor\gamma/2\rfloor}^{\gamma} b_{i,t} = \frac{(-1)^\gamma+1}{2}b_{i,\lfloor\gamma/2\rfloor} + \text{Pr}\{\hat{r}_v=1|i \text{ grešaka}, r_v=0\}, \tag{4.40}$$

gde $b_{i,t}$ predstavlja ukupnu verovatnoću da i grešaka dovodi do t pogrešnih procena bita x_v. Prema jednačini (7.8) bit x_v biće pogrešno dekodovan ako je većina njegovih procena pogrešna. Tada, za neparne vrednosti γ, samo verovatnoća da je t veće ili jednako $\lfloor\gamma/2\rfloor+1$ dovodi do greške. Ako je γ parno, tada postoji mogućnost istog broja ispravnih i pogrešnih procena.

Zaključujemo da $\gamma/2$ pogrešnih procena dovode do greške samo ako je $r_v = 1$. Verovatnoću greške pri dekodovanju moguće je dalje izraziti

$$P_v(\alpha, \epsilon) = \sum_{i=0}^{\gamma(\rho-1)} \alpha^i (1-\alpha)^{\gamma(\rho-1)-i} \left[\frac{(-1)^\gamma + 1}{2} \alpha b_{i,\lfloor \gamma/2 \rfloor} + \sum_{t=\lfloor \gamma/2 \rfloor + 1}^{\gamma} b_{i,t} \right]. \tag{4.41}$$

Koeficijente $b_{i,t}$ moguće je odrediti pronalaženjem svih uređenih particija broja i, kako je to rađeno u [117]. Ovde će biti prikazan drugačiji pristup. Uočava se da

$$\sum_{i=0}^{\gamma(\rho-1)} \alpha^i (1-\alpha)^{\gamma(\rho-1)-i} b_{i,t} \tag{4.42}$$

predstavlja ukupnu verovatnoću da je t procena bita x_v pogrešno. Procena bita će biti pogrešna ako je izračunata na osnovu neparnog broja pogrešnih bita i nije došlo do otkaza XOR kola, ili ako je broj grešaka paran, a XOR kolo je otkazalo. Tako, verovatnoća da se na izlazu j-tog XOR kolo nalazi pogrešna procena iznosi

$$P_j = \varepsilon_j (1 - A) + (1 - \varepsilon_j) A, \tag{4.43}$$

gde je

$$A = \sum_{j=0}^{\rho-1} \binom{\rho-1}{j} p^j (1-p)^{\rho-1-j} = \frac{1}{2}\big(1 - (1-2p)^{(\rho-1)}\big), \tag{4.44}$$

pri čemu desna strana jednakosti sleduje iz [22]. Svaka vrsta konfiguracione matrice $\mathbf{Q}^{i,\gamma}$ predstavlja jednu moguću kombinaciju koja dovodi do tačno i pogrešnih procena bita x_v.

Dodatak 4.B (dokaz Teoreme 4.1)

Izraz dat jednačinom (4.3) predstavlja verovatnoću pogrešnog dekodovanja proizvoljno izabranog bita za jedan scenario otkaza lokičkih kola, tj. fiksiran vektor stanja $\sigma^{(t)}$.

Pojedino stanje XOR logičkog kola $\mathbf{s}_m^{(t)}, 1 \leq m \leq \gamma$, je posledica grešaka koje u toku prenosa kroz kanal invertuju kodne bite sekvence $\mathbf{x}_{m,v}$. Broj invertovanih bita izražava se *Hamming*-ovim rastojanjem između kodne sekvence i stanja XOR kola $\mathbf{s}_m^{(t)}$. Kako su ulazi XOR kola međusobno nezavisni, verovatnoću pojave vektora stanja $\sigma^{(t)}$ moguće je odrediti množenjem pojedinačnih verovatnoća stanja XOR kola, pa se dobija

$$P\left(\sigma^{(t)}\right) = \prod_{m=1}^{\gamma} p^{d_H\left(\mathbf{s}_m^{(t)}, \mathbf{x}_{m,v}\right)} (1-p)^{M(\rho-1) - d_H\left(\mathbf{s}_m^{(t)}, \mathbf{x}_{m,v}\right)}. \tag{4.45}$$

Verovatnoća pogrešnog dekodovanja bita $x_v^{(k)}$, pod pretpostavkom da je fiksna sekvenca od M kodnih reči prenesena kroz kanal, određuje se sumiranjem proizvoda obika $P\left(\sigma^{(t)}\right) P_v(p, \varepsilon^{(t)})$ dobijenih za sve moguće vektore stanja $\epsilon^{(t)}, 1 \leq t \leq 2^{(\rho-1)\gamma M}$, dok se finalna BER vrednost izračunava dodatnim usrednjavanjem po svim kodnim bitima.

Dodatak 4.C (dokaz Leme 4.4)

Na osnovu nejednakosti (4.17) poznato je da za svako $i > 1$ važi

$$\sum_{i=2}^{\infty}(2|V_{i+1}| - K|V_i| - K|V_{i-1}|)x^i \leq 0, \tag{4.46}$$

gde je $K = 1 - 8\epsilon$. Prethodni izraz je moguće redefinisati kao

$$v(x)(2 - Kx - Kx^2) - (2|V_2| - K|V_1|)x - 2|V_1| \leq 0, \tag{4.47}$$

gde je $v(x)$ generišuća funkcija definisana na sledeći način

$$v(x) = \sum_{i=0}^{\infty}|V_{i+1}|x^i. \tag{4.48}$$

Funkciju $v(x)$ moguće je izraziti kao

$$\begin{aligned} v(x) &\leq -\frac{(2\beta - K)x + 2}{K(x_1 - x)(x_2 - x)}|V_1| = \Big[-\frac{(2\beta - K)x_2 + 2}{K(x_1 - x)(x_2 - x)} + \frac{2\beta - K}{K(x_1 - x)}\Big]|V_1| \\ &= \Big[\frac{(2\beta - K)x_2 + 2}{K(x_1 - x_2)}\Big(\sum_{i=0}^{\infty}x_1^{-i-1}x^i - \sum_{i=0}^{\infty}x_2^{-i-1}x^i\Big) + \frac{2\beta - K}{K}\sum_{i=0}^{\infty}x_1^{-i-1}x^i\Big]|V_1|, \end{aligned} \tag{4.49}$$

gde je $x_1 = -(1 + \sqrt{1 + 8/K})/2$ i $x_2 = (\sqrt{1 + 8/K} - 1)/2$. Tada sledi

$$\begin{aligned} |V_i| &\leq \Big[\frac{2 + (2\beta - K)x_2}{K(x_2 - x_1)}x_2^{-i} - \frac{2 + (2\beta - K)x_1}{K(x_2 - x_1)}x_1^{-i}\Big]|V_1| \\ &= \frac{2 + (2\beta - K)x_2}{K(x_2 - x_1)}x_2^{-i}\Big[1 - \frac{2 + (2\beta - K)x_1}{2 + (2\beta - K)x_2}\Big(\frac{x_2}{x_1}\Big)^i\Big]|V_1|. \end{aligned} \tag{4.50}$$

Kako za svako $i > 0$ i $2\beta \geq K$ važi

$$1 - \frac{2 + (2\beta - K)x_1}{2 + (2\beta - K)x_2}\Big(\frac{x_2}{x_1}\Big)^i \leq 2, \tag{4.51}$$

konačno se dobija

$$\begin{aligned} |V_i| &< \frac{4 + 2(2\beta - K)x_2}{K(x_2 - x_1)}x_2^{-i}|V_1| \\ &= \frac{4\sqrt{1 - 8\epsilon} + (2\beta - 1 + 8\epsilon)(\sqrt{9 - 8\epsilon} - \sqrt{1 - 8\epsilon})}{(1 - 8\epsilon)\sqrt{9 - 8\epsilon}}\Big(\frac{2\sqrt{1 - 8\epsilon}}{\sqrt{9 - 8\epsilon} - \sqrt{1 - 8\epsilon}}\Big)^i|V_1|. \end{aligned} \tag{4.52}$$

Poglavlje 5

Gallager B dekodovanje nepouzdanim logičkim kolima

Konstrukcijom *message-passing* iterativnog dekodera *Gallager* [26] je pokušao da ispravi nedostatke jednostavnog BF dekodera. Mana BF dekodera vezana je za pre svega za veliku korelaciju poruka koje iterativno razmenjuju čvorovi grafa. Naime, procena bita koju šalje neki varijabilni čvor u jednoj iteraciji indirektno zavisi od vrednosti koju je taj isti čvor poslao svojim susedima u prethodnoj iteraciji. Da bi umanjio sopstveni uticaj na dekodovanje, *Gallager* je predložio da varijabilni čvor v odluke donosi samo na osnovu informacija pristiglih od susednih varijabilnih čvorova, tj. čvorova iz binarnog stabla čiji je koren čvor v. Tako se uticaj korelacije "udaljio" za broj iteracija jednak $g/2$, gde je g dužina najkraćeg ciklusa *Tanner*-ovog grafa (tj. *girth*-a), kada se poruka koju je poslao čvor koji obrazuje ovaj ciklus vraća u njega. Značaj *Gallager*-ove modifikacije ogleda se u zapažanju da kada kod ne sadrži cikluse, tj. kada *Tanner*-ov graf ima oblik stabla, ne postoji korelacija poruka i tada *message-passing* dekoder ostvaruje proizvoljno malu verovatnoću greške za sve LDPC kodove iz ($\gamma \geq 3$, ρ)-regularnog ansambla [26]. Ovaj zaključak važi samo u asimptotskom slučaju, jer *Tanner*-ov graf svakog koda konačne dužine sadrži cikluse, pri čemu se g povećava logaritamski sa dužinom koda.

Pored superiornosti koju najčešće *message-passing* dekoderi ostvaruju u odnosu na BF dekodere, lakše ih je analizirati i asimptotski *density evolution* metodom [5, 108, 121]. Tragove ove analize predložio je sam *Gallager* posmatrajući *Tanner*-ov graf kao stablo, dok su kasnije Mihaljević i Golić [121] razvili sličnu metodu primenljivu u analizi kriptografskih sistema. *Richardson* i *Urbanke* [5] su dali najsistematičniji prikaz ove asimptotske metode, i pokazali da za većinu *message-passing* dekodra primenjenih na binarnom simetričnom kanalu, postoji

prag verovatnoće greške u kanalu p_t, takav da za svaku verovatnoću greške u $p < p_t$, verovatnoća zaostale greške postaje zanemarljivo mala. Pored *density evolution* tehnike, *Gallager B* dekoder moguće je analizirati i u kontekstu ekspander kodova, kako su to pokazali *Burshtein* i *Miller* [36]. Autori su dokazali da *Gallager B* dekoder, primenjen na skoro svaki kod iz ($\gamma > 5$, ρ)-regularnog ansambla, može ispraviti linearnu frakciju grešaka. S druge strane, kada je $\gamma = 3$ *Gallager B* dekoder može ispraviti samo $g/2 - 1$ grešaka ako je $g > 8$, odnosno samo dve greške kada je $g = 8$, kako je to pokazao *Chilappagari* [102].

Da se *density evolution* analiza može primeniti na *message-passing* dekodere sastavljene od nepouzdanih komponenti prvi je pokazao *Varshney* [25], koji je uočio da za razliku od pouzdanih dekodera, nepouzdani dekoderi ni asimtotski ne postižu proizvoljno malu verovatnoću greške. Navedeni zaključak važi za slučaj kada su otkazi tranzijentni, nezavisni i identično distribuirani (eng. *independent identically distributed*, i.i.d.), tj. opisuju se *von Neumann*-ovim modelem [13]. Kako je *Varshney* analizirao samo *Gallager A* dekoder, niz autora nastavio je istraživanje i pružilo slične rezultate za *Gallager B* [38, 42, 122, 123] i FAID dekodere [37,39,40,124,125]. Grupa autora predvođena *Yazdi*-jem koristila je *density evolution* analizu za određivanje optimalnog *Tanner*-ovog grafa neregularnog LDPC koda [42], dok su nebinarni LDPC kodovi posmatrani u [38]. *Huang* je analiziralo nešto složeniji model otkaza: otkazi logičkih kola su von *Neumann*-ovog tipa, dok su memorijski elementi podložni permanentnim otkazima [122], dok su *Leduc-Primeau* i *Gross* [123] predložili šemu koja kombinuje *Gallager B* dekoder sa ponavljanjem poruka. S druge strane, *Dupraz* je poboljšala notaciju *funkcionalnog praga* nepouzdanih FAID dekodera, koji je iskoristila za dizajn dekodera dobrih performansi u opsegu malih verovatnoća otkaza komponenti.

Kao što je to već napomenuto, većina istraživanja iz oblasti nepouzdanih dekodera pokazala je da su otkazi logičkih kola neželjeni i da degradiraju performanse dekodera. U ovom poglavlju biće pokazano da takvo zaključivanje nije uvek ispravo i da postoje slučajevi kada nepouzdani *Gallager B* dekoder u većem procentu ispravlja greške od dekodera koji je izgrađen od pouzdanih komponenti. Ovaj iznenađujući efekat primećen je u kodovima koji imaju male *trapping set*-ove kao što su kvazi-ciklični QC-LDPC kodovi [4]. Probabilizam inherentno prisutan u nepouzdanim logičkim kolima omogućava "razbijanje" pojedinih štetnih struktura u *Tanner*-ovom grafu i ubrzava konvergenciju ka kodnoj reči. S druge strane, u kodovima koji ne sadrže male *trapping set*-ove otkazi nastali zbog neadekvatnog vremena stabilizacije signala ne dovode do poboljšanja performansi. Takođe, u ovom poglavlju demonstrirana je i neadekvat-

nost *von Neuman*-ovog modela za analizu kodova konačene dužine, kao i mala primenljivost *density evolution* analize. Treba istaći da je problem otkaza kao rezultat smanjenja napona napajanja analizirao *Perez* [126] u kontekstu stohastičkih dekodera, ali bez konkretnog matematičkog modela otkaza, tako da simetrija dekodera nije analizirana. Rezultati prezentovani u ovom poglavlju publikovani su u [127, 128].

Ostatak poglavlja organizovan je na sledeći način. U Odeljku 5.1 opisana je ukratko arhitektura dekodera, kao i alternativni zapis GOS modela grešaka. Odeljak 5.2 posvećen je matematičkoj analizi nepouzdanog *Gallager B* dekodera, dok su performanse nekoliko tipova kodova istražene Monte Karlo simulacionim postupkom u Odeljcima 5.3 i 5.4. Zaključne napomene izložene su u Odeljku 5.5.

5.1 Arhitektura *Gallager B* dekodera i modelovanje otkaza logičkih kola

5.1.1 Arhitektura dekodera

Neka je $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ kodna reč LDPC koda poslata preko binarnog simetričnog kanala (BSC) verovatnoće greške označene sa α, gde $x_v \in \{\pm 1\}$ predstavlja polarnu vrednost bita dodeljenog čvoru v i neka je sekvenca na ulazu u *Gallager B* dekoder (izlazu iz BSC kanala) $\mathbf{r} = \{r_1, r_2, \ldots, r_n\}$. Neka su $\nu_{v,c}^{(\ell)}$ poruke koje se šalju preko grane (v, c) od varijabilnog čvora ka kontrolnom čvoru u toku ℓ-te iteracije dekodovanja. Slično, neka je $\nu_{c,v}^{(\ell)}$ poruka koja se prosleđuje preko iste grane ali u suprotnom smeru, takođe u toku ℓ-te iteracije. *Gallager B* dekoder je sumiran u nastavku.

- *Ažuriranje poruka od varijabilnih ka kontrolnim čvorovima.* Za svaki varijabilni čvor $v \in V$ važi: u toku iteracije $\ell = 0$: $\nu_{v,c}^{(0)} = r_v$, $\forall c \in \mathcal{N}_v$. Za iteracije $\ell > 0$:

$$\nu_{v,c}^{(\ell)} = \begin{cases} -r_v & \text{if } |\{c' \in \mathcal{N}_v \setminus c : \nu_{c',v}^{(\ell-1)} = -r_v\}| > \lfloor \gamma/2 \rfloor, \\ r_v & \text{inače.} \end{cases} \tag{5.1}$$

- *Ažuriranje poruka od kontrolnih ka varijabilnim čvorovima.* Za svaki kontrolni čvor $c \in C$ i $\forall v \in \mathcal{N}_c$, u toku iteracije $\ell \geq 0$:

$$\nu_{c,v}^{(\ell)} = \prod_{v' \in \mathcal{N}_c \setminus \{v\}} \nu_{v',c}^{(\ell)}. \tag{5.2}$$

Dekodovanje se prekida ako su sve provere parnosti zadovoljene ili je dostignut maksimalan broj iteracija.

Dekoder je sastavljen od *procesorskih jedinica* koji odgovaraju čvorovima u reprezentaciji dekodera bipartitnim grafom. Svaka procesorska jedinica kontrolnog čvora (eng. *check node*, CN) sastavljena je od ρ XOR logičkih kola (pri čemu svako kolo ima $\rho - 1$ ulaza) potrebnih za računanje poruka koje kontrolni čvor šalje susednim varijabilnim čvorovima. Procesorska jedinica varijabilnog čvora (eng. *variable node*, VN) računa poruke koje se šalju kontrolnim čvorovima i zahteva implementaciju γ MAJ logičkih kola sa $\gamma - 1$ ulaza [12]. Da bi dekoder uspešno radio potrebno je realizovati dodatno i logiku koja vrši finalnu procenu bita, kao i logiku za računanje provera parnosti (sindroma). Kola navedene dodatne logike moraju biti savršeno pouzdana, da bi performanse dekodera zavisile od iterativne šeme, a ne od verovatnoće otkaza ovih kola. Slično kao što je zaključeno u odeljku posvećenom BF dekoderu, ova kola se nazivaju "zlatnim kolima" i moraju biti realizovana u tehnologiji koja omogućava visok nivo njihove pouzdanosti.

Treba primetiti da je u predloženoj arhitekturi prag za odlučivanje u varijabilnim čvorovima fiksan i iznosi $\lfloor \gamma/2 \rfloor$, za razliku od orignalnog rešenja predloženog od strane *Gallager*-a koji je dozvolio adaptaciju praga u toku iteracija. Iako adaptacija praga smanjuje nivo zaostale greške u asimptotskom slučaju, zahteva implementaciju dodatne logike. Šta više, kako je to pokazao i *Gallager* adaptaciju je poželjno koristiti u nekoliko početnih iteracija, nakon čega prag konvergira upravo ka vrednosti $\lfloor \gamma/2 \rfloor$, što znači da bi se implementirana adaptacija retko koristila. Korišćenje fiksnog praga predstavlja zadovoljavajuće kompromisno rešenje između kompleksnosti i performansi. Dodatno, u ovom odeljku dominantno su posmatrani kodovi sa težinom kolana kontrolne matrice $\gamma = 3$, za koje adaptaciju nije moguće izvršiti.

5.1.2 Alternativni zapis GOS modela otkaza logičkih kola

U ovom odeljku predstavljen je alternativni zapis GOS modela grešaka, koji opisuje kombinaciona kola napajana snagom manjom od nominalne, što produžuje vreme uspostavljanja stabilnog nivoa signala [51, 54]. Ako je vreme uspostavljanja napona duže od vrednosti koja se može tolerisati dizajnom kombinacione logike, zastarela vrednost se propagira u ostatak kombinacionog kola.

Neka je $f : \{\pm 1\}^m \rightarrow \{\pm 1\}$, $m > 1$, *Boole*-ova funkcija sa m argumenata, koja u trenutku ℓ daje rezultat $z^{(\ell)} = f(y_1^{(\ell)}, y_2^{(\ell)}, \ldots, y_m^{(\ell)})$, gde su $y_1^{(\ell)}, y_2^{(\ell)}, \ldots, y_m^{(\ell)}$ argumenti funkcije

u trenutku ℓ. Zbog nepouzdanosti logičkog kola koje računa navedenu funkciju f, stvarni rezultat nije $z^{(\ell)}$ već $\mu^{(\ell)} = z^{(\ell)}e^{(\ell)}$, gde je $e^{(\ell)} \in \{\pm 1\}$ vrednost greške u trenutku ℓ. U *von Neumann*-ovom modelu otkaza vrednost $e^{(\ell)}$ je *Bernoulli*-jeva slučajna promenljiva i ne zavisi od argumenata funkcije [13].

U modelu koji opisuje propagacione greške $e^{(\ell)}$ zavisi od ulaznih argumenata, a verovatnoća da logičko kolo adekvatno ne promeni izlaznu vrednost iznosi $\Pr\{e^{(\ell)} = -1 | z^{(\ell)} \neq z^{(\ell-1)}\} = \varepsilon$, gde je $\varepsilon > 0$. S druge strane, kada je izlaz logičkog kola nepromenjen u dva susedna bitska intervala, funkcija f je uvek ispravno izračunata kao je to pretpostavljeno u [3] i [54], tj. $\Pr\{e^{(\ell)} = -1 | z^{(\ell)} = z^{(\ell-1)}\} = 0$. Naravno navedeni model je polarna predstava GOS modela otkaza opisanog u Poglavlju 2, za koji se pokazuje da osvaruje samo manju degradaciju performansi u poređenju sa znatno složenijim modelima, koji uzimaju u obzir da različite ulazne kombinacije izazivaju otkaz sa različitim verovatnoćama [3, 54].

Kao i u prethodnim poglavljima smatrano je da je vrednost ε poznata, tj. da je eksperimentalno određena simulacijama na tranzistorskom nivou ili merenjima. Zbog različite topologije, vrednost verovatnoće otkaza tipično se razlikuje za različita logička kola, implementirana u istoj tehnologiji. Ovde su sa $\varepsilon_{\oplus}$ i ε_{MAJ} označene verovatnoće otkaza XOR i MAJ logičkih kola, respektivno. Izlaz iz logičkog kola u trenutku ℓ, $\mu^{(\ell)}$, dobija se mapiranjem $\Upsilon : \{\pm 1\}^3 \to \{\pm 1\}$ na sledeći način

$$\mu^{(\ell)} = \Upsilon(z^{(\ell)}, z^{(\ell-1)}, e^{(\ell)}) = z^{(\ell)}(e^{(\ell)})^{(z^{(\ell)} - z^{(\ell-1)})/2}, \tag{5.3}$$

gde je $\Pr\{e^{(\ell)} = -1\} = \varepsilon$. Primetiti da je model otkaza redefinisan, pri čemu je uslovna verovatnoća izbačena iz notacije, a iskorišćena činjenica da je $e^{(\ell)} = 1$ sa verovatnoćom 1, kada je $z^{(\ell)} = z^{(\ell-1)}$.

Dodatno, smatrano je da su otkazi nekog logičkog kola nezavisni od otkaza drugih logičkih kola. Primetiti da predloženi model predstavlja generalizaciju "žičanog" modela opisanog u [25], gde su samo *von Neumann*-ovi otkazi razmatrani.

5.2 Analiza nepouzdanog *Gallager B* algoritma dekodovanja

Model otkaza predstavljen u Odeljku 5.1.2 ponaša se kao binarni kanal sa memorijom. U nastvaku će biti objašnjeno kako memorija utiče da performanse *Gallager B* dekodera postanu zavisne od redosleda kodnih reči koje se dekoduju. Neka je primljeni vektor izražen kao $\mathbf{r} = \mathbf{x} \cdot \mathbf{n}$, gde "$\cdot$" označava operaciju množenja element po element (eng. *pointwise multiplication*)

kodnog vektora $\mathbf{x}$ i vektora šuma $\mathbf{n}$. Neka su $\nu_{v,c}^{(-1)}$ i $\nu_{c,v}^{(-1)}$ izlazi iz funkcija za ažuriranje poruka koje se razmenjuju između varijabilnog čvora v i kontrolnog čvora c u trenutku neposredno pre inicijalne iteracije kodne reči koja se trenutno dekoduje. To su u stvari poruke poslate u poslednjoj iteraciji prethodno dekodovane kodne reči. Teorema data u nastavku definiše potreban uslov da verovatnoća uspešnog dekodovanja ne zavisi od poslate kodne reči.

Teorema 5.1. *Verovatnoća zaostale greške (bilo po bitu ili okviru) Gallager B dekodera u prisustvu otkaza u logičkim kolima nezavisna je od poslate kodne reči* $\mathbf{x}$ *ako i samo ako* $\nu_{v,c}^{(-1)} = x_v A_v$ *i* $\nu_{c,v}^{(-1)} = x_v B_v$, $\forall v \in V$ *i* $\forall c \in C$, *gde su* $A_v, B_v \in \{\pm 1\}$.

Dokaz: Neka su $\mu_{c,v}^{(\ell)}(\mathbf{r})$ i $\mu_{v,c}^{(\ell)}(\mathbf{r})$, respektivno, poruke poslate od kontrolnog čvora c ka varijabilnom čvoru v i od varijabilnog čvoru v ka kontrolnom čvoru c, u toku iteracije ℓ, za fiksni primljeni vektor $\mathbf{r}$. Ove vrednosti su dobijene prema jednačini (5.3) na osnovu ispravno izračunatih poruka $\nu_{c,v}^{(\ell)}(\mathbf{r})$ i $\nu_{v,c}^{(\ell)}(\mathbf{r})$ i odgovarajućih sekvenci grešaka koje opisuju otkaze kola $e_{c,v}^{(\ell)}$ i $e_{v,c}^{(\ell)}$.

Neka je pretpostavljeno da važi $\nu_{v,c}^{(-1)} = x_v A_v$ i $\nu_{c,v}^{(-1)} = x_v B_v$, za neke proizvoljno izabrane $A_v, B_v \in \{\pm 1\}$. Iskaz dat u formulaciji teoreme biće dokazan matematičkom indukcijom. Na osnovu poznate simetrije operacija u varijabilnim čvorovima u toku iteracije $\ell = 0$ imamo $\nu_{v,c}^{(0)}(\mathbf{r}) = \nu_{v,c}^{(0)}(\mathbf{x} \cdot \mathbf{n}) = x_v \nu_{v,c}^{(0)}(\mathbf{n})$ [25]. Kako su ove vrednosti primljene iz kanala (nisu izračunate kombinatornom logikom), one su prosleđene bez grešaka susednim kontrolnim čvorovima. Na osnovu simetrije operacija u kontrolnim čvorovima sledi $\nu_{c,v}^{(0)}(\mathbf{r}) = x_v \nu_{c,v}^{(0)}(\mathbf{n})$ [25], i dobija se

$$\begin{aligned} \mu_{c,v}^{(0)}(\mathbf{r}) &= \Upsilon(\nu_{c,v}^{(0)}(\mathbf{r}), x_v B_v, e_{c,v}^{(0)}) \\ &= x_v \nu_{c,v}^{(0)}(\mathbf{n}) (e_{c,v}^{(0)})^{x_v(\nu_{c,v}^{(0)}(\mathbf{n}) - B_v)/2} \\ &= x_v \Upsilon(\nu_{c,v}^{(0)}(\mathbf{n}), B_v, e_{c,v}^{(0)}) = x_v \mu_{c,v}^{(0)}(\mathbf{n}). \end{aligned} \tag{5.4}$$

Slično se zaključuje da važi $\mu_{v,c}^{(1)}(\mathbf{r}) = x_v \Upsilon(\nu_{v,c}^{(1)}(\mathbf{n}), A_v, e_{v,c}^{(1)}) = x_v \mu_{v,c}^{(1)}(\mathbf{n})$.

Neka je pretpostavljeno da važi $\mu_{v,c}^{(\ell)}(\mathbf{r}) = x_v \mu_{v,c}^{(\ell)}(\mathbf{n})$, $\forall v \in V$, $\forall c \in C$ i $\ell > 1$. Na osnovu činjenice da važi $\prod_{v:\mathcal{N}(c)} x_v = 1$ i simetrije operacija u kontrolnim čvorovima, opisane u [25], sledi $\nu_{c,v}^{(\ell)}(\mathbf{r}) = x_v \nu_{c,v}^{(\ell)}(\mathbf{n})$. Tada, slično kao što je to izvedeno u jednačini (5.4), dobijamo

$$\mu_{c,v}^{(\ell)}(\mathbf{r}) = \Upsilon(x_v \nu_{c,v}^{(\ell)}(\mathbf{n}), x_v \nu_{c,v}^{(\ell-1)}(\mathbf{n}), e_{c,v}^{(\ell)}) = x_v \mu_{c,v}^{(\ell)}(\mathbf{n}). \tag{5.5}$$

Dodatno, koristeći opet činjenicu da su operacije u varijabilnim čvorovima simetrične i primenjujući jednačinu (5.3), dobija se $\mu_{v,c}^{(\ell+1)}(\mathbf{r}) = x_v \mu_{v,c}^{(\ell+1)}(\mathbf{n})$. Kako su sve poruke razmenjene

između varijabilnog čvora v i njegovih susednih kontrolnih čvorova jednake proizvodu x_v i odgovarajuće poruke kada je primljen samo vektor $\mathbf{n}$, performanse dekodera su nezavisne od poslate kodne reči.

S druge strane, kada uslov dat u formulaciji teoreme nije zadovoljen, odnosno ako je $\nu_{v,c}^{(-1)} \neq x_v A_v$ ili $\nu_{c,v}^{(-1)} \neq x_v B_v$, tada sledi da $\mu_{v,c}^{(1)}(\mathbf{r}) \neq x_v \mu_{v,c}^{(1)}(\mathbf{n})$. Kako poruke $\mu_{v,c}^{(1)}$ propagiraju u naredne iteracije, dekodovana vrednost, a samim tim i verovatnoća greške, zavisi od $\mathbf{x}$. ■

Uslovi potrebni da bi verovatnoća zaostale greške bila nezavisna od $\mathbf{x}$ su krajnje nerealni. Ti uslovi mogu biti zadovoljeni samo podešavajući inicijalna stanja logičkih kola prema kodnoj reči koja se dekoduje, a koja je nepoznata dekoderu. Tako zaključujemo da je u slučaju korelisanih otkaza logičkih kola korektivna sposobnost dekodera uvek uslovna i zavisi od $\mathbf{x}$.

S druge strane, teorema pokazuje da je takvo ponašanje dekodera povezano sa činjenicom da nije moguće obezbediti poštovanje uslova u toku prve iteracije dekodovanja. Ako u toku prve iteracije ne bi dolazilo do otkaza logičkih kola, uslovi simetrije bi bili zadovoljeni. Savršeno pouzdanu prvu iteracija moguće je postići usporavajući dekodovanje dozvoljavajući stabilizaciju signala na izlazu iz logičkih kola. Kako je frekvencija sistemskog takta manja ne dolazi do otkaza kao posledice prevelikog propagacionog kašnjena. Dekoder kod koga je prva iteracija pouzdana u nastavku će se nazivati *modifikovani Gallager B dekoder*.

5.3 Performanse kodova bez malih *trapping set*-ova

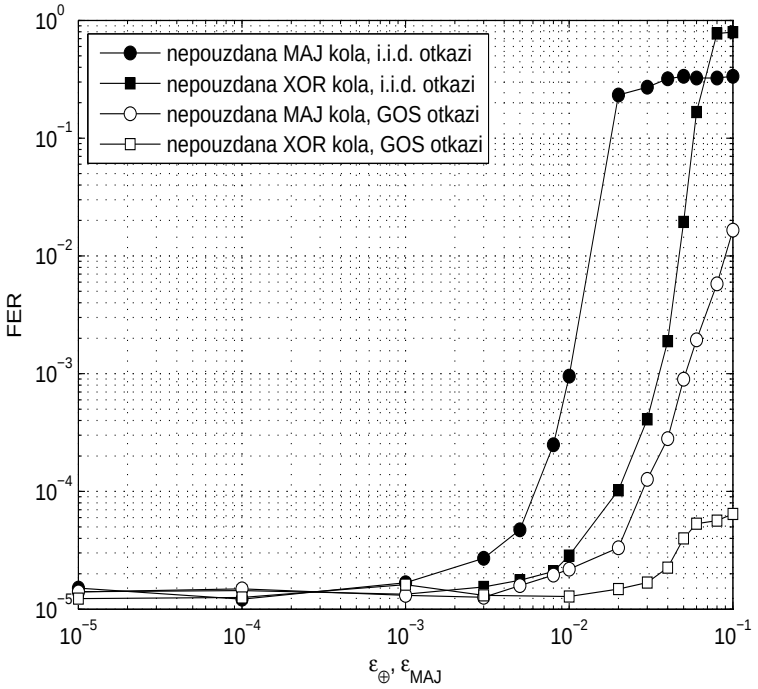
Kako je to ranije napomenuto, relevantni rad vezan za nepouzdani *Gallager B* dekoder, u većem obimu obuhvata asimptotsku *density evolution* analizu, u prisustvu i.i.d. otkaza opisanih *von Neumann*-ovim modelom [38, 122, 123]. U prethodnom odeljku pokazano je da kada su otkazi logičkih kola vremenski korelisani, dekoder nije simetričan, pa samim tim nije moguće primeniti ovu asimptotsku metodu. S druge strane, modifikovani *Gallager B* dekoder zadovoljava uslov simetrije i moguće ga je ispitivati asimptotski. Prisustvo otkaza transformiše pouzdani dekoder bez memorije u dekoder sa memorijom, čak i kada u *Tanner*-ovom grafu ne postoje ciklusi. *Density evolution* dekodera sa memorijom analizirali su skorije *Janulewicz* i *Banihashemi* [129]. Iako su autori uspeli da međusobnu zavisnost poruka koje se razmenjuju između čvorova grafa opišu *Bayes*-ovom mrežom, pokazali su da računarska kompleksnost analize raste eksponencijalno sa brojem iteracija, čak i kada memorija postoji samo u porukama

koje napuštaju varijabilne čvorove, a ne i kontrolne čvorove. Dodatno, ni aproksimacija predložena u [129] ne dovodi do računarski efikasne *density evolution* analize, koja ostaje otvoren problem. Zbog toga je u ovom poglavlju fokus na praktično upotrebljivim kodovima konačne dužine, čije performanse su ispitivane Monte Karlo simulacionim postupkom, gde je broj iteracija dekodovanja limitiran na 100. Analizirani su kodovi koji ne sadrže male *trapping set*-ove kao što su, na primer, kodovi konstruisani na bazi latinskih kvadrata (LS-LDPC) kodovi [78], ili kodovi baszirani na progresivnom rastu grana grafa (PEG-LDPC) kodovi [77]. Da bi se ilustrovala zavisnost performansi dekodera od kodnih reči koje se prenose tri simulaciona moda su korišćena:

- mod M_0: samo se prenosi kodna reč sastavljena od svih jedinica;
- mod M_H: naizmenično se prenose dve kodne reči čije je međusobno *Hamming*-ovo rastojanje veliko;
- mod M_R: prenose se slučajno izabrane kodne reči.

Na slici 5.1 ilustrovane su performanse dekodera za slučaj koda u oznaci LS(155,64) [78] procenjene za dva tipa otkaza u logičkim kolima, kada je najrealističniji mod M_R podrazumevan. Reč je o kodu iz (3,5)-regularnog ansambla pri čemu je $|V| = 155$ i $|C| = 93$. Dva scenarija su od interesa: (i) kada su operacije u kontrolnim čvorovima pouzdane, a samo MAJ logička kola nepouzdana, i (ii) kada su operacije u varijabilnim kolima pouzdane, a samo XOR logička kola otkazuju. Treba istaći da degradacija performansi nije uočena za verovatnoće otkaza niže od 10^{-3}, bez obzira na posmatrani model otkaza logičkih kola, što je i zaključak analize OS-MAJ dekodera iz Poglavlja 4. Međutim, kada otkazi logičkih kola postanu češći dostiže se prag nakon koga verovatnoća greške po okviru (FER) ubrzano raste. Vrednost praga zavisi od tipa logičkih kola korišćenih u dekoderu, i uočava da MAJ kola imaju niži prag, pa je obezbeđivanje njihove pouzdanosti značajnije.

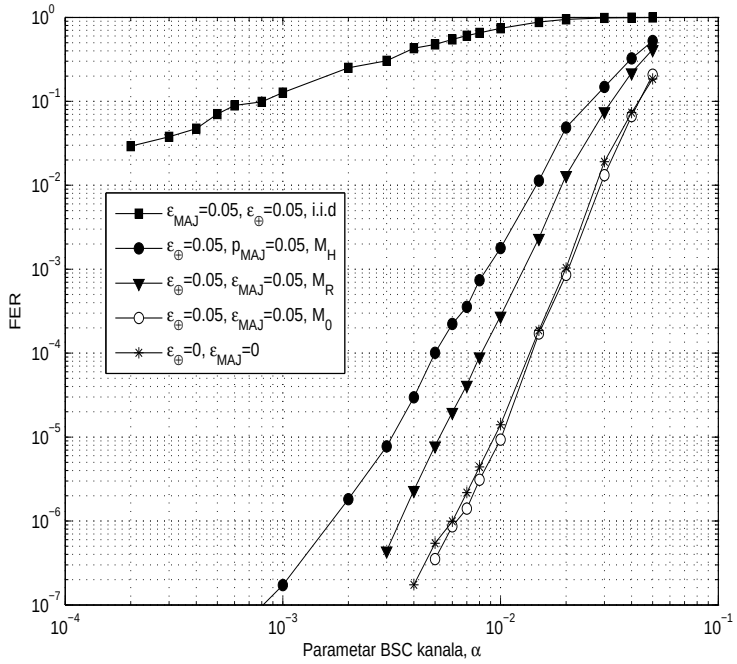
GOS model grešaka svodi se na *von Neumann*-ov model kada je logičko kolo konstantno u "lošem stanju", tj. menja izlaznu vrednost u svakom bitskom intervalu. U tom smislu, *von Neumann*-ov model se može smatrati suviše pesimističkim i varljivim, što je poosebno uočljivo u regionu sa visokom nepouzdanošću XOR logičkih kola. Upotreba GOS modela otkriva veću robusnost na otkaze XOR kola nastale usled smanjenja snage napajanja. Na primer, čak i za ekstremno veliku verovatnoću otkaza u lošem stanju od $\varepsilon_{\oplus} = 0.1$ dekoder zadržava nisku



Slika 5.1: Poređenje *von Neumann*-ovog (i.i.d.) i GOS modela grešaka za kod LS(155,64) (α=0.01).

verovatnoću zaostale greške. S druge strane, prema *von Neumann*-ovom modelu, nepouzdani dekoder u proseku nadmaši i nekodovani sistem.

Za umerene ili niske vrednosti verovatnoće greške u kanalu, u većini slučajeva samo će nekoliko od 155 kodnih bita biti pogrešno primljeno, pa će većina poruka $\nu_{v,c}^{(0)}$ predstavljati ispravne procene kodnih bita. Ako je u toku prve iteracije frekvencija otkaza takođe niska, većina poruka poslatih iz varijabilnih čvorova će ostati nepromenjena. S druge strane, ako broj otkaza nije zanemarljiv, nakon prve iteracije, broj pogrešnih procena značajno će se povećati. Uticaj koju prva iteracija dekodovanja ima na ukupne performanse dekodera ilustrovana je na slici 5.2. Moguće je primetiti da ovi otkazi logičkih kola dominantno određuju korektivnu sposobnost dekodera. Kada se prenosi samo kodna reč svih jedinica (mod M_0) ne dolazi do degradacije performansi u poređenju sa pouzdanim dekoderom, čak i za visoke verovatnoće otkaza ($\varepsilon_{\oplus} = \varepsilon_{MAJ} = 0.05$). S druge strane, simulacioni mod M_H otkriva da najgori slučaj izaziva degradaciju verovatnoće greške od nekoliko redova veličine. Značajno lošije performanse postiže i najrealističniji scenario, koji podrazumeva slanje slučajno izabranih kodnih reči.

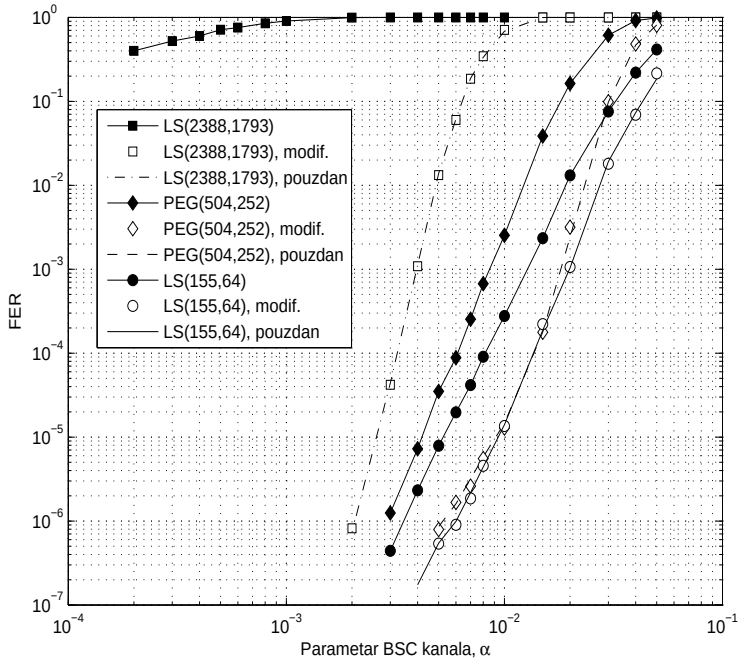


Slika 5.2: Zavisnost performansi dekodera od redosleda kodnih reči za LS(155,64) LDPC kod.

Slika 5.3 ilustruje poboljšanje koje postiže modifikovani *Gallager B* dekoder, u kome se operacije u prvoj iteraciji obavljaju savršeno pouzdano, za nekoliko kodova težine kolona kontrolne matrice $\gamma = 3$. Iako je napravljen od nepouzdanih komponenti, modifikovani *Gallager B* dekoder postiže nivo zaostale greške kao savršeno pouzdani dekoder. S druge strane, performanse nepouzdanog *Gallager B* dekodera bez modifikacije lošije su za nekoliko redova veličine. Na primer, može se primetiti da je kod LS(2388,1793) neupotrebljiv za širok opseg verovatnoće greške u kanalu, kada se ne koristi predložena modifikacija.

5.4 Performanse kodova sa malim *trapping set*-ovima

U ovom odeljku biće analizirani QC-LDPC kodovi, koje je predložio *Tanner* u svom značajnom članku [4]. Prednost QC-LDPC kodova ogleda se u jednostavnoj implementaciji procesa kodovanja, koja s druge strane za posledicu ima formiranje štetnih struktura u *Tanner*-ovom grafu, zvanih *trapping set*-ovi. Postojanje *trapping set*-ova u nekom kodu determiniše performanse koda u regionu male verovatnoće greške u kanalu. Poznato je da QC-LDPC kodovi sadrže (5,3) *trapping set*, prikazan na slici 5.4), koji onemogućava *Gallager*-B algoritam da ispravlja sve trostruke greške [78]. Pri tome crni krugovi odgovaraju bitima koji su pogrešno

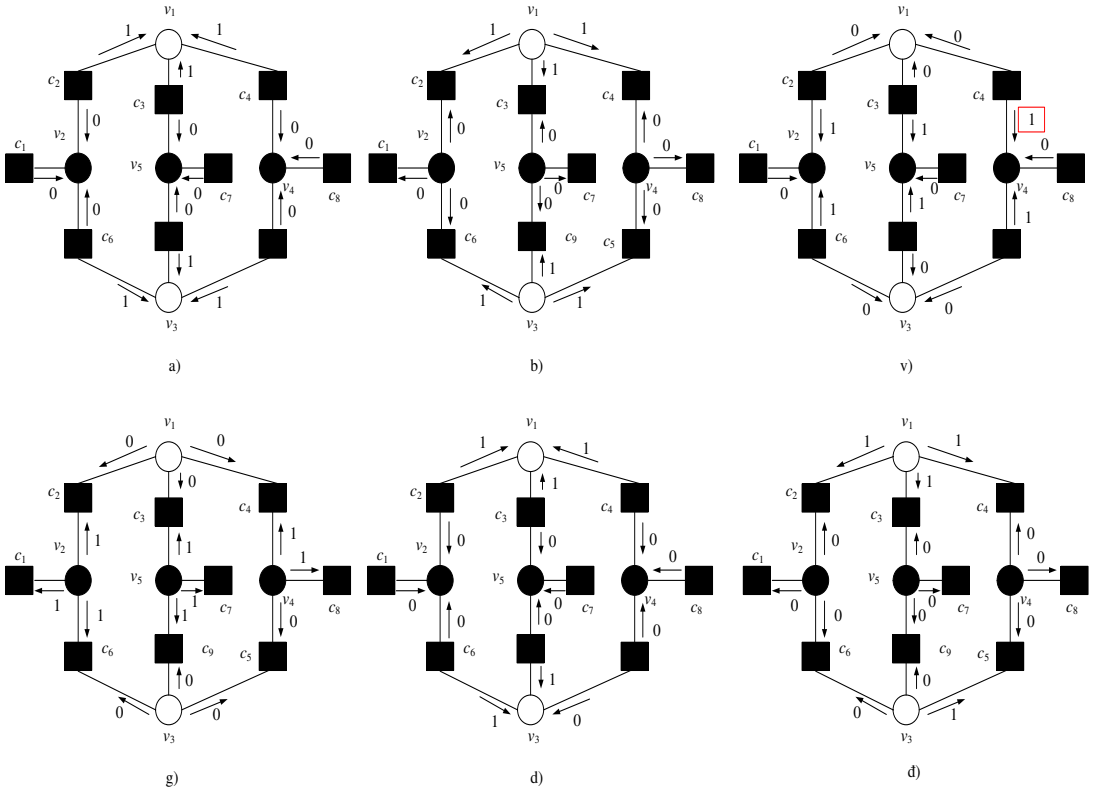


Slika 5.3: Poređenje različitih kodova za slučaj simulacionog moda M_R ($\varepsilon_{\oplus} = \varepsilon_{MAJ} = 0.05$).

primljeni, dok beli krugovi označavaju isprtavno primljene bite. Beli kvadrati opisuju kontrolne čvorove koji odgovaraju zadovoljenim proverama parnosti, dok crni kvadrati opisuju nezadovoljene provere parnosti.

Podgraf koji odgovara (5,3) *trapping set*-u u oznaci $G' = (V' \cup C', E')$, gde je $V' = \{v_1, v_2, v_3, v_4, v_5\}$, a $C' = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9\}$, pri čemu su pogrešno primljeni biti koji odgovaraju čvorovima v_2, v_4 i v_5. Na slici 5.4.a označene su vrednosti poruka koje kontrolni čvorovi šalju svojim susedima u toku prve iteracije. Topologija grafa G' diktira da se nakon prve iteracije (slika 5.4.b) pogrešne vrednosti u čvorovima isprave, ali se greške unose na do sada ispravne čvorove v_1 i v_3. U narednoj iteraciji (slika 5.4.c) odluke se invertuju, pa procene inicijalno pogrešnih bita v_2, v_4 i v_5 postaju ponovo pogrešne. Dinamika dekodovanja osciluje, pri tom se ni u jednom trenutku kodna reč ispravno ne dekoduje. S druge strane, otkazi u logičkim kolima mogu da naruše oscilovanje procesa dekodovanja i dovedu do ispravljanja ovog obrasca grešaka neispravljivog za dekoder sastavljen od pouzdanih komponenata. U analizi koja sledi smatrano je da su XOR kola nepouzdana, dok se operacije u varijabilnim čvorovima smatraju savršeno pouzdanim.

Nakon prve iteracije, koja prema Teoremi 5.1 mora biti savršeno pouzdana, XOR kola koja su promenila izlaznu vrednost mogu otkazati, pri čemu neće svaki otkaz dovesti do ispravl-



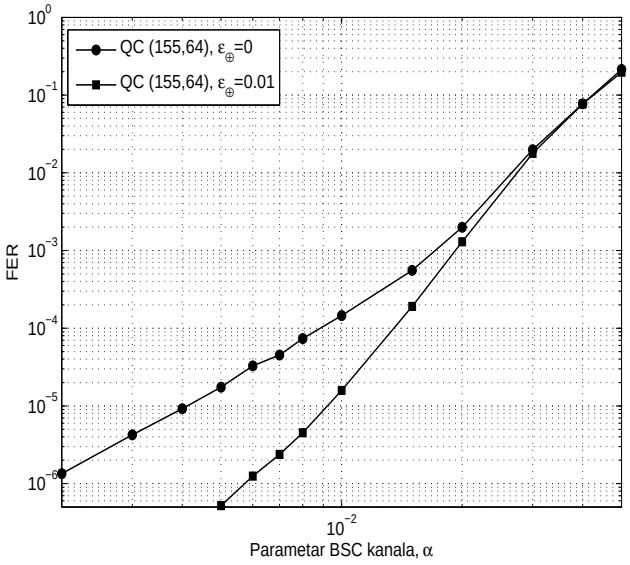
Slika 5.4: Dekodovanje (5,3) *trapping set*-a: a)-v) bez otkaza u logičkim kolima; g)-đ) sa otkazom u XOR logičkim kolima.

janja kodnih bita. Može se uočiti da će do ispravljanja doći ako se invertuje jedna od poruka koju kontrolni čvorovi prosleđuju nekom od inicijalno pogrešnih varijabilnih čvorova. Takvih poruka je $N_f = 6$ od ukupno $N_t = 14$ poruka koje prema GOS modelu grešaka mogu dovesti do otkaza nekog logičkog XOR kola. Verovatnoća da se izađe iz *trapping set*-a može se grubo proceniti kao

$$P_{br} \approx N_f \varepsilon_{\oplus} (1 - \varepsilon_{\oplus})^{N_t - 1}, \tag{5.6}$$

gde, na primer, ako je $\epsilon_{\oplus} = 0,015$ dobijamo $P_{br} \approx 0,154$. Pod navedenim uslovima u proseku oscilatorni proces se prekida otkazima XOR kola svakih sedam iteracija dekodovanja. Neka je na primer došlo do otkaza XOR kola koje računa poruku koju čvor c_4 šalje varijabilnom čvoru v_4 (označena crvenom bojom na slici 5.4.v). Tada se menja struktura poruka koje se šalju iz varijabilnog čvora v_4 u narednoj iteraciji (slika 5.4.g), tako da jednačina provera parnosti c_5 postaje zadovoljena, a procene bita v_4 ostaju ispravne i u narednoj iteraciji. U sledećoj iteraciji ispravlja se i varijabilni čvor v_5 (slike 5.4.d i 5.4.đ), dok se i preostali pogrešan čvor v_1 ispravlja u toku još jedne dodatne iteracije.

Sumirano, dovoljnan je otkaz samo jednog XOR kola da se ispravi posmatrana trostruka

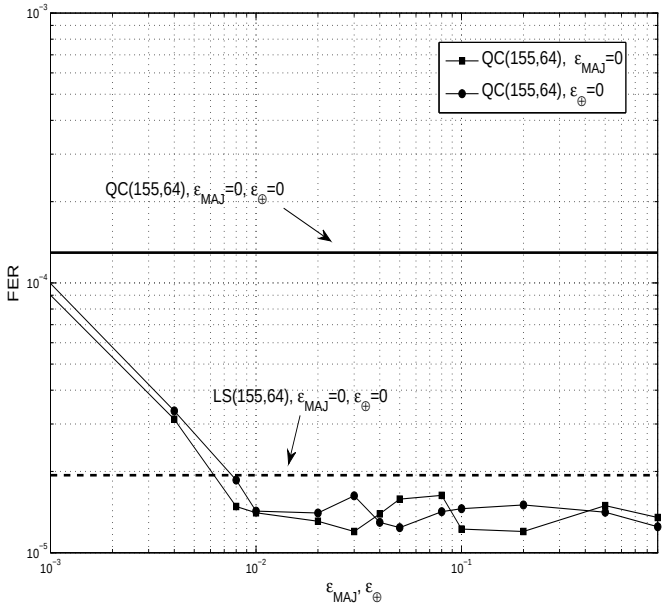


Slika 5.5: Performanse QC(155,64) koda.

greška, neotklonjiva pouzdanim *Gallager B* dekoderom, što se u zavisnosti od nivoa nepouzdanosti hardvera može desiti za svega nekoliko iteracija (u posmatranom slučaju u proseku će se za samo 10 iteracija ispraviti obrazac greške). Pri tom treba uočiti da prilikom dekodovanja *trapping set*-a većina poruka u *Tanner*-ovom grafu ne menja vrednost u toku iteracija, pa neće biti podložna propagacionim greškama u logičkim kolima. Na taj način nepouzdanost deluje samo onda kada postoji problematična struktura, dok se poruke koje se razmenjuju između ispravno primljenih kodnih bita prenose bez grešaka.

Analiza predhodnog obrasca grešaka ilustruje loše performanse koje postižu QC-LDPC kodovi dekodovani pouzdanim *Gallager B* dekoderom. Kako dekoder ne može ispraviti sve trostruke greške, verovatnoća zaostale greške je relativno visoka, kao što je to prikazano na slici 5.5, gde su ilustrovane performanse koda QC(155,64). Ovaj kod ima dužinu 155 bita, kodni količnik blizak 0,42, minimalno *Hamming*-ovo rastojanje $d_{min} = 20$, pri čemu su konstrukcioni parametri $\gamma = 3$, $\rho = 5$. U regionu verovatnoće greške u kanalu $\alpha < 0,01$, verovatnoća zaostale greške je ograničena verovatnoćom pojave neotklonjivog obrasca datog na slici 5.4. S druge strane, ovakve sekvence grešaka moguće je ispraviti ako su XOR kola inherentno nepouzdana, pa za isti broj iteracija (L=100) nepouzdani dekoder premašuje pouzdani za više od jednog reda veličine. Tako na primer kada je $\alpha = 0,005$ pouzdani dekoder ostvaruje verovatnoću greške od 2×10^{-5}, dok pod uticajem otkaza logičkih kola vrednost zaostale greške dostiže 5×10^{-7}.

Na slici 5.6 ispitan je uticaj nivoa nepouzdanosti logičkih kola na primećeno poboljšanje

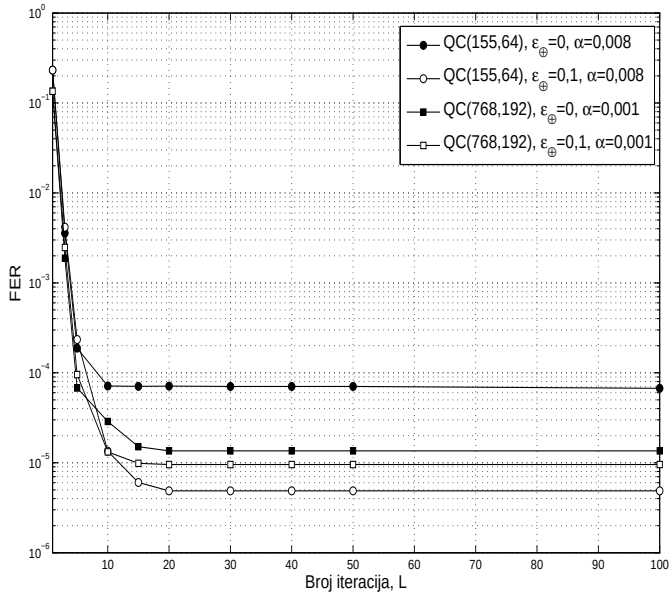


Slika 5.6: Uticaj različitih nivoa nepouzdanosti na performanse QC(155,64) koda ($\alpha = 0,01$).

performansi QC(155,64) koda. Posebno je interesantna činjenica da povećanje nepouzdanosti uvek vodi ka nižoj verovatnoći greške, pa se najveće poboljšanje ostvaruje i za ekstremno nepouzdana logička kola. Ovaj primer je još jedna ilustracija činjenice da se poruke koje se razmenjuju između čvorova grafa u proseku retko menjaju, a da je njihova promena često indikacija postojanja *trapping set*-a. Tada otkaz kola (sa koliko god velikom verovatnoćom !) dovodi do napuštanja *trapping set*-a i usmerava konvergenciju procesa dekodovanja ka korektnoj kodnoj reči. Da nepouzdani (modifikovani) *Gallager B* dekoder zaista ispravlja sve (5,3) *trapping set*-ove u QC(155,64) kodu, potvrđeno je njegovim poređenjem sa LS(155,64) kodom čije performanse su dostignute dopuštanjem logičkim kolima da rade nepouzdano.

Posmatranjem slike 5.7 primećuje se takođe da hardverski otkazi ne utiču negativno na vreme konvergencije iterativnog *Gallager B* dekodera. Jedna od glavnih prednosti *Gallager B* dekodera (pored jednostavnosti) je mali broj iteracija potreban za dostizanje maksimalno mogućeg nivo greške. Pokazuje se da je u većini slučajeva dovoljno iterirati 20 puta, nakon čega se greške mogu proglasiti neotklonjivim. Iako nepouzdani dekoder ispravlja i neke neotkonjive greške pouzdanog dekodera, za to mu često ne treba veći broj iteracija, kako je to prikazano na slici 5.7 na primerima dva QC koda. Brza konvergencija nepouzdanog *Gallager B* dekodera iskorišćena je pri dizajniranju hibridnog dekodera u Poglavlju 6.

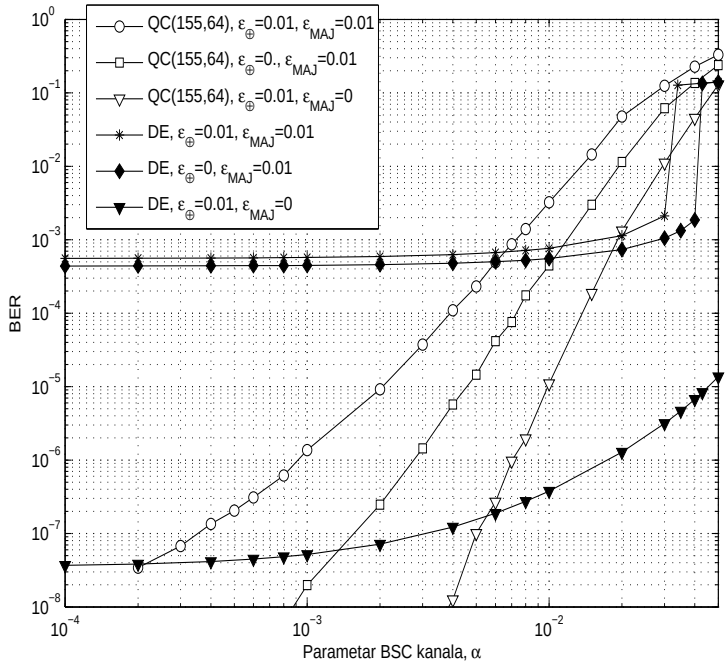
Treba primetiti da je u prethodno opisanoj analizi podrazumevano da su finalne procene



Slika 5.7: Uticaj broja iteracija na performanse QC(155,64) i QC(768,192) kodova.

bita kao i računanje sindroma savršeno pouzdani. Implementacija ovih "zlatnih" logičkih kola omogućava identifikaciju pronalaska kodne reči (sindrom jednak vektoru svih nula) i završetak dekodovanja onda kada je kodna reč uspešno pronađena. Ako ove operacije ne bi bile pouzdane dešavalo bi se da dekoder ispravno dekoduje kodnu reč, dok njegove odluke ne bi bile ispravno interpretirane, a samim tim ne bi dovele do poboljšanja performansi. Značaj "zlatnih" logičkih kola ilustrovan je na slici 5.8, gde su poređene performanse koda konačne dužine QC(155,64) kod, koga su operacije računanja sindroma pouzdane, i rezultata *density evolution* (DE) asimptotske tehnike, dobijene na osnovu [38]. Iako su rezultati dati za *von Neumann*-ov tip otkaza, zaključci adekvatno ilustruju značaj pouzdanog računanja sindroma.

Posmatrana su tri slučaja i to: kada su nepouzdana samo XOR logička kola, samo MAJ logička kola, kao i kada su sve poruke koje razmenjuju čvorovi *Tanner*-ovog grafa nepouzdani. Treba naglasiti da ovde nije u fokusu potencijalno poboljšanje performansi usled otkaza logičkih kola, pa je vrednost verovatnoće otkaza izabrana tako da samo pokaže generalni trend ponašanja dekodera. Uočljivo je da čak kada su i XOR i MAJ logička kola nepozdana verovatnoća zaostale greške se smanjuje skoro linearno sa smanjenjem parametra kanala α. S druge strane za beskonačno dugačak kod, čiji graf ne sadrži cikluse, verovatnoća greške brzo konvergira ka vrlo visokoj vrednosti i sugeriše da kodovi iz $(3, 5)$-regularnog ansambla nisu upotrebljivi, pri zadatim uslovima otkaza logičkih kola. Kako u *density evolution* tehniku nije moguće



Slika 5.8: Performanse QC(155,64) koda i asimptotski dobijene vrednosti za *von Neumann*-ov modela grešaka.

uvrstiti računanje sindroma, ona postaje neadekvatna za analizu praktičnih sistema baziranih na kodovima konačne dužine. Suprotno od slučaja pouzdanih dekodera, gde je DE analiza bila suviše optimistična, obećavajući proizvoljno malu verovatnoću greške, mana DE tehnike primenjene na nepouzdane dekodere ogleda se u činjenici da potcenjuje njihove performanse, posebno u regionu niskih verovatnoća greške u kanalu.

5.5 Zaključak

U ovom poglavlju istražene su osobine *Gallager B* dekodera u prisustvu vremenski korelisanih otkaza logičkih kola. Pokazano je da degradacija performansi usled smanjenja napona napajanja logičkih kola zavisi od redosleda kodnih reči, što je prvi takav zaključak u kontekstu nepouzdanih *message-passing* dekodera. Dodatno, otkriven je razlog zbog koga simetrija dekodera nije ispunjena i performanse nisu nezavisne od kodnih reči. Na osnovu matematički izvedenog uslova, modifikovan je originalni dekoder tako da degradacija performansi bude zanemarljivo mala, kako su to pokazale simulacije više praktično značajnih kodova koji ne sadrže male *trapping set*-ove. Modifikovani dekoder je pokazao ne samo da je otporan na uticaj

otkaza u logičkim kolima, već i da postoje slučajevi kada otkazi mogu dovesti do superiornijih performansi.

Istraživanje izloženo u ovom poglavlju otvara nekoliko novih pravaca istraživanja. Poznato je da iterativni dekoderi LDPC kodova ne mogu dostići *Shannon*-ov kapacitet, kada se otkazi opisuju *von Neumann*-ovim modelom [25, 38]. Suprotno od toga, simulaciona analiza prezentovana u ovom poglavlju sugeriše optimističnije rezultate za vremenski korelisane otkaze. Od posebnog interesa je i dalja karakterizacija efekata koji dovode do željene stohastičke rezonance dekodera.

Poglavlje 6

Iterativni dekoder na bazi agregacije poruka

Značajan broj iterativnih dekodera LDPC kodova predložen je u protekle dve decenije. Poznato je da SPA dekoder [26] ostvaruje dobre performanse na binarnim simetričnim kanalima. Međutim, velika kompleksnost SPA dekodera onemogućava njegovu primenu u većem broju praktičnih realizacija, od *flash* memorija, do optičkih sistema. Nekoliko kvantizovanih MP dekodera predloženo je sa ciljem da se ubrza proces dekodovanja, a da pritom korektivne sposobnosti budu uporedive sa sposobnostima SPA dekodera [92, 93]. Efekti kvantizacije su najprimetniji prilikom dekodovanja 3-levo-regularnih kodova, što dovodi do visokog nivoa greške u *error-floor* regionu, kao u slučaju *min-sum* dekodera. Drugi kvantizovani FAID dekoderi, predloženi u [94, 95], ostvaruju performanse bolje od performansi SPA, za određen broj praktično značajnih kodova. Na žalost, kompleksnost FAID dekodera i dalje je značajno veća u poređenju sa dekoderima koji propagiraju tvrde odluke.

Da bi se popunila praznina između jednostavnih BF ili *Gallager B* dekodera i FAID dekodera, skorije je predloženo nekoliko interesantnih rešenje. *Nguyen* i *Vasić* [88] su konstruisali klasu TBBF algoritama u kojima su poruke, koje se razmenjuju između čvorova *Tanner*-ovog grafa, ojačane dodatnim bitom, što povećava korektivnu sposobnost 3-levo-regularnih kodova. Dodatno, isti autori su razvili i okvir *združene korecije grešaka* u kome paralelan rad više komplementarnih TBBF dekodera sa velikom verovatnoćom ispravljaju sve četvorostruke greške. S druge strane, sličan koncept nazvan MUDRI razvio je i Ivaniš [110] u kome se predlaže višestruka upotreba PGDBF dekodera. *Mobini* [130] je konstruisao MP dekoder koji ažurira meku informaciju primljenu iz kanala na osnovu binarnih poruka koje se razmenjuju preko

grana *Tanner*-ovog grafa. *Sassatelli* [131] je dizajnirala *two-bit* MP dekoder za koji je dokazala garantovano ispravljanje svih trostrukih grešaka na 4-levo-regularnim kodovima.

U ovom poglalvju predložen je novi BF metod dekodovanja LDPC kodova, u kome pored poruka koje se standardno razmenjuju između čvorova *Tanner*-ovog grafa, varijabilni čvorovi komuniciraju direktno. U predloženom pristupu varijabilni čvorovi sakupljaju informacije sa većeg dela grafa u toku jedne iteracije, za razliku od standardnog pristupa u kome jednobitske poruke propagiraju u više iteracija. Pokazano je da ako varijablni čvorovi upoznaju svoje okruženje (u smislu primljenih vrednosti, a ne topografije grafa), sa velikom verovatnoćom se eliminišu *trapping set*-ovi male težine. Takođe, pokazano je da je kompleksnost predloženog rešenja uporediva sa kompleksnošću jednostavnog *Gallager A/B* dekodera. Kao poseban segment predložen je i hibridni dekoder, koji kombinuje predloženi algoritam i nepouzdani *Gallager B* dekoder opisan u Poglavlju 5. Hibridni dekoder je jednostavniji od združene korekcije TBBF dekodera ili MUDRI strategije, gde samo veliki broj iteracija dovodi do poboljšanja. Kompleksnost dekodera predloženog u ovom radu samo je dva puta veća od kompleksnosti *Gallager B* dekodera, ali njegove performanse prevazile sve praktično značajne dekodere, koji donose tvrde odluke na 3-levo-regularnim kodovima. Rezultati prezentovani u ovom poglavlju objavljeni su u [132].

Ostatak poglavlja organizovan je kako sledi. U Odeljku 6.1 opisan je novi pristup razbijanja *trapping set*-ova. Odeljak 6.2 posvećen je hibridnom dekoderu, a takođe sadrži i analizu korektivnih sposobnosti dekodera. Numerički rezultati dati su u Odeljku 6.3, dok je analiza kompleksnosti predloženog rešenja data u Odeljku 6.4. Kratka zaključna diskusija prezentovana je u Odeljku 6.5.

6.1 Novi algoritam za razbijanje *trapping set*-ova

6.1.1 Opis algoritma

Dekoderi koji rade na bazi tvrdih odluka su jednostavni, ali ostvaruju visok nivo greške u *error-floor* regionu zbog postojanja određenih struktura u *Tanner*-ovom grafu, što ima za posledicu da specifične konfiguracije grešaka male težine sprečavaju konvergenciju dekodera ka koreknoj kodnoj reči. Osnovu problema čini zapažanje da varijabilni čvorovi ne poznaju dovoljno dobro svoje okruženje, pa na bazi pogrešnih impresija donose loše odluke. U nastavku će biti pokazano kako inkorporiranje čak i dela znanja koje imaju susedni varijabilni čvorovi dovodi

do značajnog poboljšanja performansi. Konstruisani dekoder nazvan je MAE (eng. *Message Aggregation Enhanced*) dekoderom.

Neka je $U(v_i)$ broj nezadovoljenih kontrolnih čvorova u susedstvu varijabilnog čvora v_i, i neka su sa V_C i $V_{\bar{C}}$ označeni skupovi varijabilnih čvorova povezani samo na zadovoljene i nezadovoljene kontrolne čvorove, respektivno, odnosno

$$V_C = \{v_i \in V | U(v_i) = 0\} \tag{6.1}$$

i

$$V_{\bar{C}} = \{v_i \in V | U(v_i) = \gamma\}. \tag{6.2}$$

Kao meru zadovoljenosti određenih provera parnosti definisana je i sledeća kriterijumska funkcija

$$\psi(v_i) = 1 - \mathbb{1}_{V_C}, \tag{6.3}$$

gde je sa $\mathbb{1}$ označena indikatorska funkcija.

Vrednost $\psi(v_i) = 0$ je indikator da je (na osnovu znanja susednih kontrolnih čvorova) vrednost varijabilnog čvora v_i "verovatno ispravna". Slično, kontrolni čvor c_m okružen samo sa verovatno ispravnim varijabilnim čvorovima je "verovatno zadovoljen". Skup ovakvih kontrolnih čvorova može se označiti sa

$$\mathcal{C}_C = \{c_m \in C | \sum_{v_k \in \mathcal{N}(c_m)} \psi(v_k) = 0\}. \tag{6.4}$$

Svaka iteracija sastoji se iz tri koraka. U prvom koraku, na osnovu vrednosti kriterijumske funkcije ψ, identifikuju se verovatno ispravni čvorovi, što izoluje preostale čvorove nazvane "potencijalno pogrešnim", čiji skup označavamo sa V_I. Cilj algoritma je da iterativno smanjuje kardinalni broj skupa potencijalno pogrešnih čvorova, dok u njemu ne ostanu samo zaista pogrešni varijabilni čvorovi. Slično, kako su varijabilni čvorovi iz $V_{\bar{C}}$ povezani samo sa nezadovoljenim kontrolnim čvorovima, to su pogrešni sa velikom verovatnoćom. U prvom koraku vrednost svih takvih čvorova treba invertovati.

U drugom koraku, da bi se dalje smanjio skup potencijalno pogrešnih čvorova V_I, koji nakon prvog koraka sadrži čvorove v_i povezane sa $0 < U(v_i) < \gamma$ nezadovoljenih kontrolnih čvorova, algoritam uzima u obzir binarna stabla sa korenima u tim čvorovima, označena sa $\mathcal{T}_i$ (pogledati Odeljak 3.3), agregira poruke na nivou binarnih stabala i inkorporira ih u pravilo invertovanja. Vrednosti $\psi(v_i)$ se ponovo izračunavaju i prosleđuju direktno svim čvorovima

iz $\mathcal{T}_i$. Ako varijabilni čvor primi nule od svih varijabilnih čvorova sa kojima deli kontrolne čvorove, napušta skup potencijalno pogrešnih čvorova. Dalje, u trećem koraku invertuje se vrednost potencijalno pogrešnog čvora v_i ako i samo ako je povezan na nezadovoljeni kontrolni čvor c_j i ne postoji ni jedan drugi varijabilni čvor povezan na c_j koji pripada skupu potencijalno pogrešnih čvorova.

Treba primetiti da je kriterijum invertovanja restriktivan i dozvoljava invertovanje pogrešnih vrednosti sa velikom verovatnoćom, dok je verovatnoća da se ispravna vrednost invertuje zanemarljiva. Opisani algoritam može se formalno matematički opisati kako je to da prikazano u nastavku.

Neka je dato binarno stablo $\mathcal{T}_i$. Neka $\mathcal{T}_{i\setminus j}$ označava podgraf koji isključuje deo binarnog stabla povezanog na kontrolni čvor c_j, odnosno,

$$V_{\mathcal{T}_{i\setminus j}} = V_{\mathcal{T}_i} \setminus \mathcal{N}(c_j), \tag{6.5}$$

$$C_{\mathcal{T}_{i\setminus j}} = C_{\mathcal{T}_i} \setminus c_j. \tag{6.6}$$

Primetiti da je čvor v_i takođe isključen iz $\mathcal{T}_{i\setminus j}$. Svakom podgrafu $\mathcal{T}_{i\setminus j}$ pridružena je kriterijumska funkcija $\Upsilon : \{0,1\}^{\rho-1} \rightarrow \{0,1\}$ definisana na sledeći način

$$\Upsilon(\mathcal{T}_{i\setminus j}) = \begin{cases} 0, & \text{if } \exists c_m \in C_{\mathcal{T}_{i\setminus j}} \cap \mathcal{C}_C \\ 1, & \text{inače .} \end{cases} \tag{6.7}$$

Vrednost $\Upsilon(\mathcal{T}_{i\setminus j}) = 0$ govori da se v_i isključuje iz V_I, dok u suprotnom ovaj čvor ostaje u V_I. Međutim, da li će vrednost v_i iz V_I zaista biti invertovana zavisi i od drugih varijabilnih čvorova povezanih sa c_j, kao što je to već napomenuto. Algoritam za razbijanje *trapping set*-ova formalno je definisan u nastavku.

Algoritam razbijanja *trapping set*-ova

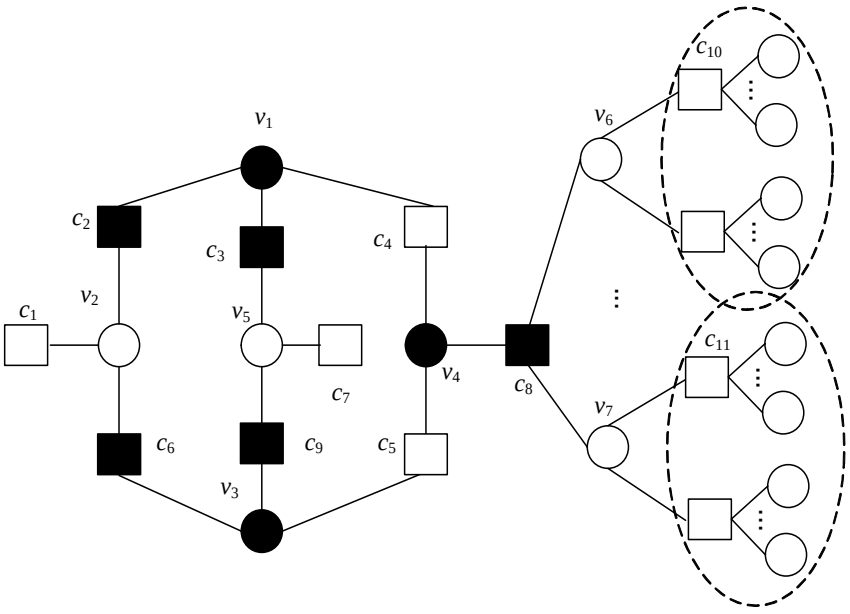
1) Invertovati vrednost svakog čvora povezanog sa γ nezadovoljenih kontrolnih čvorova.

2) Izračunati $\psi(v_i)$ i $\Upsilon(\mathcal{T}_{i\setminus j})$, $\forall v_i \in V$ i $\forall c_j \in \mathcal{N}(v_i)$.

3) Invertovati vrednost v_i ako $\exists c_j \in \mathcal{N}(v_i)$ takvo da je $c_j = 1$, $\Upsilon(\mathcal{T}_{i\setminus j}) = 1$ i $\forall v_m \in \{\mathcal{N}(c_j) \setminus v_i\}$ $\Upsilon(\mathcal{T}_{m\setminus j}) = 0$.

4) Ponavljati prethodna tri koraka dok svi kontrolni čvorovi ne postanu zadovoljeni.

Predloženi algoritam je u nastavku objašnjen na primeru pojave trostruke greške na (3,5)-regulanom kodu. Podgraf na kome je uneta greška predstavljen je na slici 6.1, gde crni odnosno



Slika 6.1: Podgraf koji odgovara unosu trostruke greške, korišćen u primeru.

beli krugovi označavaju pogrešno primljene i ispravne bite, respektivno, dok crni i beli kvadrati, predstavljaju nezadovoljene i zadovoljene kontrolne čvorove, respektivno. Podgraf od interesa označen sa $G' = (V' \cup C', E')$ sadrži skup varijabilnih čvorova $V' = \{v_1, v_2, \ldots, v_5\}$ i skup kontrolnih čvorova $C' = \{c_1, c_2, \ldots, c_9\}$. Neka se smatra da je podgraf *izolovan* u smislu da ne postoji varijabilni čvor u binarnim stablima čvorova $\mathcal{N}(C') \setminus V'$ koji deli kontrolni čvor sa čvorovima iz V'. Uvedena pretpostavka biće formalizovana kasnije. Neka se posmatra kontrolni čvor $c_8 = 1$, povezan na pogrešno primljeni čvor v_4. Može se uočiti da je $\Upsilon(\mathcal{T}_{4\setminus 8}) = 1$, jer su c_4 i c_5 povezani sa čvorovima koji učestvuju u nezadovoljenim proverama parnosti. Međutim, može se primetiti da je $\Upsilon(\mathcal{T}_{6\setminus 8}) = 0$ i $\Upsilon(\mathcal{T}_{7\setminus 8}) = 0$, jer su c_{10} i c_{11} povezani sa čvorovima čiji su svi susedni kontrolni čvorovi zadovoljeni. Slični argumenti mogu se primeniti za druge varijabilne čvorove povezane sa c_8, što daje indikaciju da je potrebno invertovati vrednost v_4. Primenjujući isti princip na kontrolne čvorove c_2 i c_6 može se zaključiti da jedino varijabilni čvorovi v_1 i v_3 ne pružaju dokaz o svojoj korektnosti, pa ih treba invertovati.

Primetiti da je u prethodnom primeru uspešno dekodovanju u samo jednoj iteraciji bilo moguće zbog pretpostavke o izolovanost *trapping set*-a. Međutim, moguće je pokazati da je dovoljan uslov da bi greške bile ispravljene samo izolovanost kontrolnog čvora c_8. Kasnije će biti definisan manje restriktivan uslov izolovanosti, koji je značajan za analizu korektive sposobnosti algoritma datog u sledećem odeljku.

6.1.2 Implementacija MAE dekodera na bazi tvrdih odluka

Kao što je to napomenuto u Odeljku 3.3 iterativni dekoder razmenuje poruke preko grana *Tanner*-ovog grafa. Neka je sa $\mu_e^{(\ell)}$ označena poruka koja se šalje preko grane $e = (v_i, c_j)$ od varijabilnog čvora v_i ka kontrolnom čvoru c_j u ℓ-toj iteraciji. Slično, $\nu_e^{(\ell)}$ označava poruku poslatu preko iste grane u suprotnom smeru (od kontrolnog čvora c_j ka varijabilnom čvoru v_i) takođe u toku ℓ-te iteracije. Vrednosti $\mu_e^{(\ell)}$ dobijaju se mapiranjem $\Phi^{(\ell)} : \{0,1\}^{\gamma+1} \to \{0,1\}$, odnosno, $\mu_e^{(\ell)} = \Phi^{(\ell)}(\boldsymbol{\nu}_i^{(\ell)}, y_i)$, gde je $\boldsymbol{\nu}_i^{(\ell)} = (\nu_e^{(\ell)})_{e\in\mathcal{E}(v_i)}$, dok se $\Psi^{(\ell)} : \{0,1\}^{\rho} \to \{0,1\}$ koristi za računanje $\nu_e^{(\ell)}$ kao $\nu_e^{(\ell)} = \Psi^{(\ell)}(\boldsymbol{\mu}_i^{(\ell-1)})$, gde je $\boldsymbol{\mu}_i^{(\ell-1)} = (\mu_e^{(\ell-1)})_{e\in\mathcal{E}(c_j)}$.

U ovom delu odeljka biće pokazano kako se predloženi algoritam može lako implementirati u formi standardnog dekodera na bazi tvrdih odluka, koji podrazuveva da se u jednoj iteraciji preko grana *Tanner*-ovog grafa razmenjuju samo binarne poruke. To omogućava da se računanje kriterijumske funkcije obavlja lokalno u varijabilnim čvorovima, a da nisu potrebne neke globalne operacije. Posledično, broj iteracija se povećava, jer se odluke o invertovanju prolongiraju dok sve vrednosti kriterijumskih funkcija ne propagiraju do željenih varijabilnih čvorova. Primećuje se da se poruke koje napuštaju čvorove *Tanner*-ovog grafa računaju različito u različitim iteracijama. Prvi korak predloženog algoritma može se obaviti u jednoj iteraciji, dok su potrebne tri dodatne iteracije za drugi i treći korak – jedna za računanje $\psi(v_i)$, druga za određivanje $\Upsilon(\mathcal{T}_{i\setminus j})$ za svaki par (v_i, c_j), i treća za propagiranje $\Upsilon(\mathcal{T}_{i\setminus j})$ ka svim susednim varijabilnim čvorovima. Međutim, sve navedene operacije mogu se obaviti preko dve *Boole*-ove funkcije argumenata $\mathbf{p} = (p_1, \ldots, p_A)$ and $\mathbf{q} = (q_1, \ldots, q_B)$, definisane na sledeći način

$$F(\mathbf{p}, \mathbf{q}) = \prod_{i=1}^{A} p_i \oplus \bigoplus_{i=1}^{B} q_i, \tag{6.8}$$

$$G(\mathbf{p}, \mathbf{q}) = \prod_{i=1}^{A} p_i \times \bigoplus_{i=1}^{B} q_i, \tag{6.9}$$

gde $\times$ označava AND operaciju. Poruke od varijabilnog čvora v_i ka kontrolnom čvoru c_j preko grane e određuje se kao $\mu_e^{(\ell)} = \Phi^{(\ell)}(\boldsymbol{\nu}_i^{(\ell)}) = F(\mathbf{p}_e^{(\ell)}, \mathbf{q}_e^{(\ell)})$, gde je $\mathbf{p}_e^{(\ell)} = \boldsymbol{\nu}_i^{(\ell)}$, i

$$\mathbf{q}_e^{(\ell)} = \begin{cases} \{\hat{x}_i\}, & \ell = 0 \bmod 4, \\ \emptyset, & \ell = 1 \bmod 4, \\ \{\nu_e^{(\ell)}, 1\}, & \ell = 2 \bmod 4, \\ \{\hat{x}_i, 1\}, & \text{inače} \end{cases} \tag{6.10}$$

Treba naglasiti da $\hat{x}_i$ predstavlja trenutnu procenu i-tog kodnog bita. Slično, poruke poslate preko grane e u suprotnom smeru se računaju kao

$$\nu_e^{(\ell)} = \Psi^{(\ell)}(\boldsymbol{\mu}_i^{(\ell-1)}) = \begin{cases} F(\emptyset, \tilde{\mathbf{p}}_e^{(\ell)}), & \ell = 0 \bmod 4, \\ F(\tilde{\mathbf{p}}_e^{(\ell)}, \tilde{\mathbf{q}}_e^{(\ell)}), & \ell = 1 \bmod 4, \\ G(\{\tilde{\mathbf{p}}_e^{(\ell)} \setminus e\}, \tilde{\mathbf{q}}_e^{(\ell)}), & \text{inače}, \end{cases} \tag{6.11}$$

gde je $\tilde{\mathbf{p}}_e^{(\ell)} = \boldsymbol{\mu}_i^{(\ell-1)}$, a $\tilde{\mathbf{q}}_e^{(\ell)} = \{\mu_e^{(\ell-1)}, 1\}$. Može se primetiti da četiri uzastopne iteracije formiraju ciklus i da se invertovanje bita obavlja u toku prve i poslednje iteracije ciklusa. U ostalnim iteracijama čvorovi razmenujuju vrednosti koje se odnose samo na broj nezadovljenih provera parnosti. Na kraju svakog ciklusa dekodovanje se restartuje i procene kodnih bita $\hat{x}_i$, $1 \leq i \leq n$, dobijene na kraju ciklusa predstavljaju ulaznu informaciju u naredni ciklus. Tako zaključujemo da je dekoder simetričan i da njegove operacije ne zavise od prenesenih kodnih reči. Zbog kompletnosti izlaganja formalni opis implementacije MAE dekodera dat je u nastavku.

Algoritam 1: Implementacija MAE dekodera na bazi tvrdih odluka

Input: $\mathbf{y} = (y_1, y_2, \ldots, y_n)$

$\ell \leftarrow 1$

$\hat{\mathbf{x}} \leftarrow \mathbf{y}, \forall v_i \in V : \mu_e^{(0)} \leftarrow y_i, \forall e \in \mathcal{E}(v_i)$

$\mathbf{s} \leftarrow \hat{\mathbf{x}}\mathbf{H}^T(\forall c_j \in C : s_j \leftarrow F(\emptyset, \tilde{\mathbf{p}}_e^{(0)}))$

while $\mathbf{s} \neq \mathbf{0}$ **and** $\ell < L$ **do**

$\forall c_j \in C : \nu_e^{(\ell)} \leftarrow \Psi^{(\ell)}(\boldsymbol{\mu}_i^{(\ell-1)}), \forall e \in \mathcal{E}(c_j)$

$\forall v_i \in V : \mu_e^{(\ell)} \leftarrow \Phi^{(\ell)}(\boldsymbol{\nu}_i^{(\ell)}), \forall e \in \mathcal{E}(v_i)$

if $\ell = 0 \bmod 4$ **then**

$\hat{x}_i \leftarrow F(\mathbf{p}_e^{(\ell)}, \{\hat{x}_i\}), 1 \leq i \leq n$

else if $\ell = 3 \bmod 4$ **then**

$\hat{x}_i \leftarrow F(\mathbf{p}_e^{(\ell)}, \{\hat{x}_i, 1\}), 1 \leq i \leq n$

end if

$\mathbf{s} \leftarrow \hat{\mathbf{x}}\mathbf{H}^T(\forall c_j \in C : s_j \leftarrow F(\emptyset, \tilde{\mathbf{p}}_e^{(\ell)}))$

$\ell \leftarrow \ell + 1$

end while

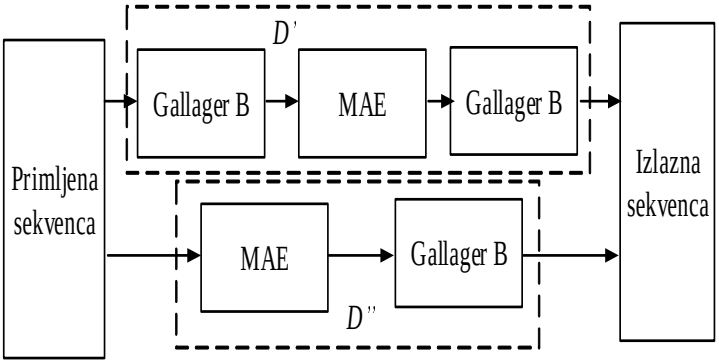
Output: $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)$

6.2 Hibridni dekoder sa agregacijom poruka i analiza korekcionih sposobnosti

Može se primetiti da ispravljanje pogrešno primljenog bita u predstavljenom algoritmu zavisi od ispravnosti susednih varijabilnih čvorova – varijabilnih čvorova sa kojima čvor deli susede u *Tanner*-ovog grafu. Ako je broj nezadovoljenih kontrolnih čvorova nizak greške je moguće ispraviti, ali algoritam neće ispraviti greške koje uzrokuju pojavu većeg broja nezadovoljenih kontrolnih čvorova. Na osnovu toga zaključujemo da će algoritam postizati dobre performanse u *error-floor* regionu. Ranjivost koju algoritam pokazuje u regionu kada su greške u kanalu češće moguće je umanjiti nekim drugim iterativnim dekoderom. Tako je predložen hibridni dekoder D_1 sastavljen od dekodera D' i D'', koji porocesiraju paralelno, kako je prikazano na slici 6.2.

- D' **dekoder**. Primljena reč iz kanala se prvo dekoduje nepouzdanim *Gallager B* dekoderom (opisanim u Poglavlju 5). Ako se dekodovanje ne obavi uspešno, procenjene vrednosti bita koriste se kao ulaz MAE dekodera. Dodatni *Gallager B* dekoder se koristi za ispravljanje preostalih grešaka.

- D'' **dekoder**. Dekodovanje se započinje MAE dekoderom, a sve neispravljene greške prosleđuju se nepouzdanom *Gallager B* dekoderu.



Slika 6.2: Blok šema hibridnog dekodera D_1.

Dekodovanje označeno sa D' oslanja se na činjenicu da će se u toku svake iteracije broj grešaka smanjiti. Međutim, postoje štetne strukture koje izazivaju oscilaciju dinamike dekodovanja *Gallager B* dekodera, pa se broj pogrešnih bita može i povećati. Zbog toga je dotat dekoder D'' u kome dekodovanje kreće od MAE algoritma. Treba primetiti da je u praksi dovoljno im-

plementirati samo po jedan *Gallager B* i MAE blok, kako se prostorni diverziti može zameniti vremenskim.

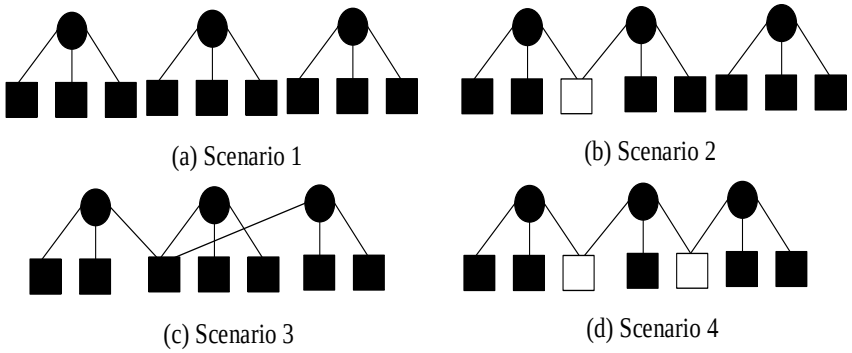
U nastavku odeljka ispitane su korektivne sposobnosti hibridnog dekodera, koje se oslanjaju na uslov izolovanosti definisan u nastavku.

Definicija 6.1. *Neka je dat podgraf* $G' = (V' \cup C', E')$, $G' \subset G$, *i skup* $C'' \subset C'$, *gde je* $C'' = \{c_j \in C' | c_j = 1\}$. *Za* G' *se kaže da je izolovan ako je* $\forall c_j \in C''$ *takvo da* $\forall v_i \in \{\mathcal{N}(c_j) \setminus V'\}$ $\exists c_m \in \{\mathcal{N}(v_i) \setminus c_j\} \subset \mathcal{C}_C$.

Prezentovana strategija dekodovanja primenjena na neki podgraf koji sadrži čvorove iz G' uključuje i čvorove izvan G' u proces donošenja odluka o čvorovima iz V'. Uslov izolovanosti garantuje da ti čvorovi neće biti prepreka ispravljanju grešaka. Pri tome se ne zahteva kompletna izolazija podgrafa, kako se to najčešće razmatra u literaturi, već se samo zabranjuju pojedine strukture, što je znato manje restriktivan uslov. U stavu koji sledi ispitane su korektivne sposobnosti dekodera D''.

Stav 6.1. *MAE dekoder primenjen na 3-levo regularni kod, čiji Tanner-ov graf* G *ima girth* $g = 8$, *može da ispravi sve trostruke greške, čiji podgrafovi zadovoljavaju uslov izolovanosti iz Definicije 7.1, dok* D'' *može ispraviti sve trostruke greške bez obraničenja.*

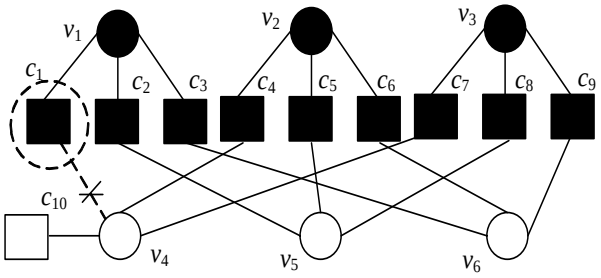
Skica dokaza: Na slici 6.3 su predstavljene sve četiri moguće konfiruracije trostrukih grešaka u kodu *Tanner*-ovog grafa, za koji važi $g = 8$. U nastavku dokaza sva četiri slučaja biće analizirana pojedinačno, slično kao što je to rađeno u [131, 133]. Na slici 6.4 ilustrovan je najgori



Slika 6.3: Četiri moguće konfiruacije trostrukih grešaka u grafu sa *girth*-om $g = 8$.

slučaj koji odgovara Scenariju 1. Greške su na čvorovima v_1,v_2 i v_3 koji ne dele susedne kontrolne čvorove, što znači na će biti invertovani u toku prvog koraka MAE dekodera. Međutim, ispravne vrednosti bita v_5 i v_6 će takođe biti invertovane. Prema uslovu izolovanosti

pretpostavka da je da ne postoji sused čvora c_1 koji je povezan na nezadovoljene kontrolne čvorove, što znači da v_4 ostaje ispravan nakon prvog koraka. Sada su nezadovljeni samo kontrolni čvorovi c_2, c_3, c_5, c_6, c_8 i c_9. Lako se uočava da ovi kontrolni čvorovi ne dele susede u grafu za koji važi $g = 8$, pa će v_5 i v_6 biti ispravljeni u toku sledeća dva koraka MAE dekodera. Primetiti da je navedenu konfiguraciju greške moguće ispraviti i samo nepouzdanim *Gallager B* dekoderom.



Slika 6.4: Podgraf koji odgovara Scenariju 1 sa slike 6.3.

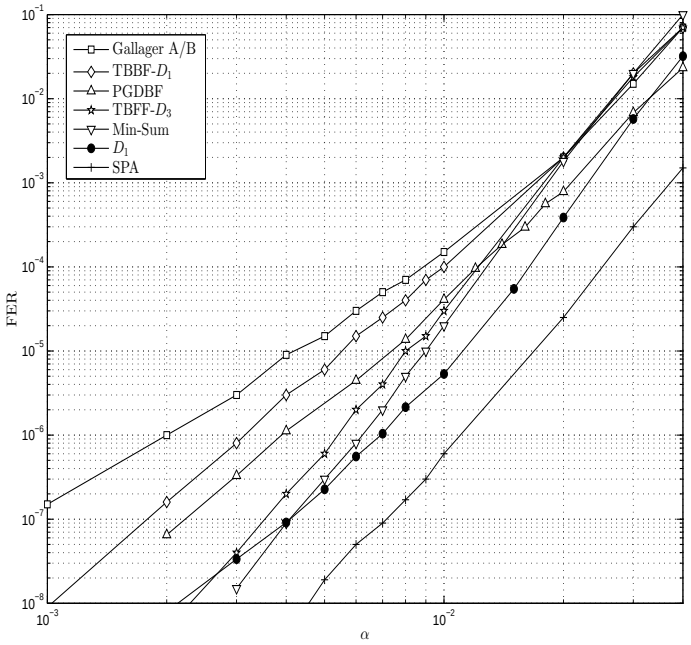
U Scenariju 2 ne postoje dva pogrešna varijabilna čvora koji dele kontrolni čvor, pa ne postoji varijabilni čvor koji je povezan sa sva tri nezadovoljena kontrolna čvora. To znači da ni jedan čvor neće biti invertovan u toku prvog koraka MAE strategije. Ako je *trapping set* izolovan tada postoji kontrolni čvor $c_j = 1$ i varijabilni čvor v_i takav da važi $\Upsilon(\mathcal{T}_{i\setminus j}) = 1$. Ovo dovodi do ispravljanja varijabilog čvora povezanog na c_j. Preostale greške biće ispravljene u sledećoj iteraciji. Primetiti da iako *trapping set* i nije izolovan, preostale dve greške je moguće ispraviti *Gallager B* dekoderom.

U scenariju 3 sva tri pogrešna varijabilna čvora dele kontrolni čvor, što znači da ne postoji varijabilni čvor u *Tanner*-ovom grafu, povezan sa tri nezadovoljena kontrolna čvora. Prema pravilu MAE dekodera sve greške će biti ispravljene u toku prvog koraka.

Scenario 4 već je prikazan na slici 6.1. To je poznati (5,3){2} *trapping set* [78]. Primena MAE algoritma dovodi do ispravljanja sve tri greške ako je podgraf izolovan. Takođe greške je moguće ipraviti i nepouzdanim *Gallager B* dekoderom, ako graf nije izolovan. ■

6.3 Numerički rezultati

U ovom odeljku određene su performanse dekodera D_1, predloženog u odeljku 6.2, izražene preko verovatnoće greške po okviru (FER) za nekoliko značajnih 3-levo-regularnih kodova.

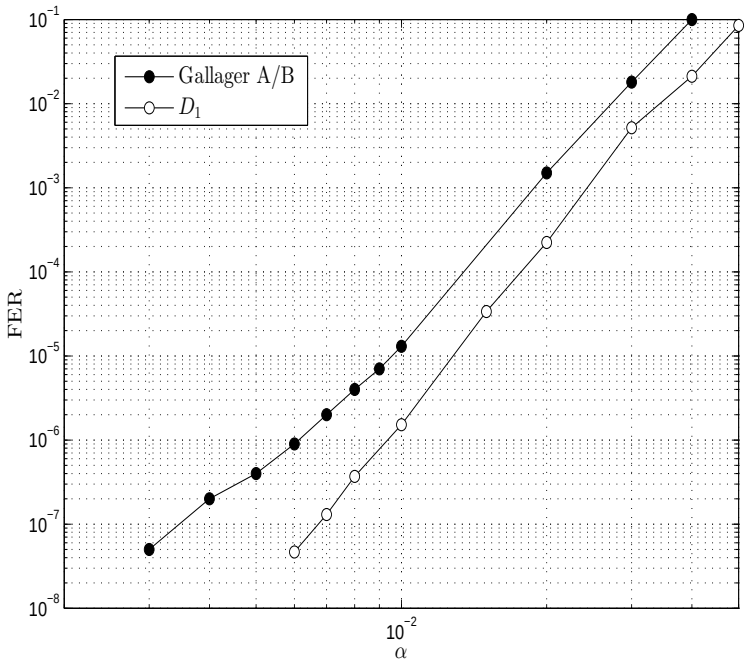


Slika 6.5: FER performanse različitih dekodera na *Tanner*-ovom QC(155,64) kodu.

Segmenti nepouzdanog *Gallager A/B* dekodera rade po 20 iteracija, dok MAE segmenti iteriraju 16 puta, što daje ukupno 92 iteracija D_1 dekodera. *Min-sum*, PGDBF i SPA dekoderi se uključuju po 100 iteracija. Maksimalan broj iteracija TBBF-D_1 dekodera je 30. Kako TBBF-D_3 dekoder predstavlja paralelnu konkatenaciju četiri TBBF dekodera, ukupno radi $4 \times 30 = 120$ iteracija.

Na slici 6.5 poređene su FER performanse različitih dekodera na popularnom *Tanner*-ovom QC(155,64) kodu. Može se primeti da ako je ciljana vrednost FER viša ili jednaka 10^{-7} dekoder D_1 postiže bolje performanse od svih razmatranih rešenja osim SPA. Na primer, D_1 dekoder ostvaruje FER od 10^{-6} kada je verovatnoća greške u BSC kanalu $\alpha = 0.007$, dok na primer TBBF-D_1 dekoder, koji takođe može da ispravi sve trostruke greške, isti nivo greške postiže samo ako je $\alpha < 0.003$. Treba naglasiti da superiornost dekodera D_1 nema negativnu stranu izraženu kroz preveliku komplesnost, jer je njegova realizacija značajno manje zahtevna od svih razmatranih rešenja osim *Gallager A/B* dekodera. Kompleksnost dekodera posebno je razmatrana u narednom odeljku.

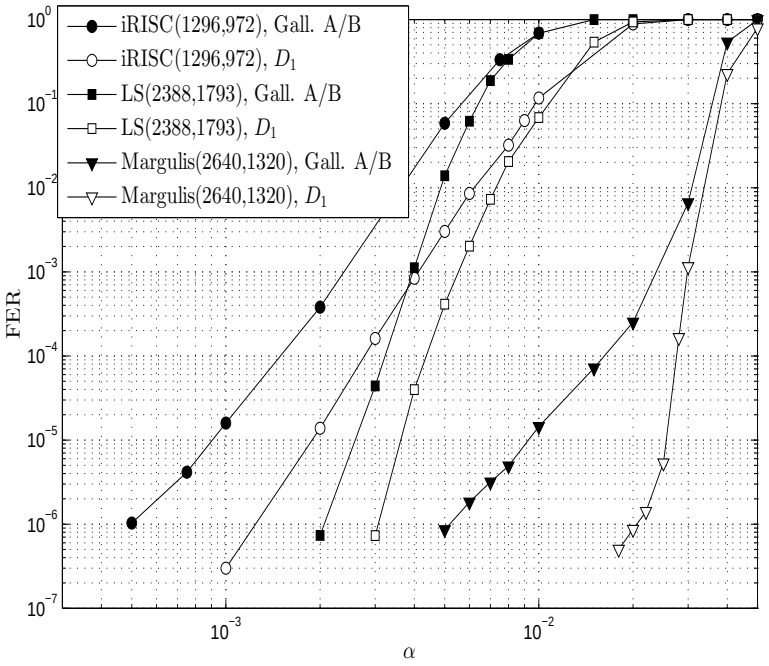
Poznato je da pouzdani *Gallager A/B* ostvaruje loše performanse na *Tanner*-ovom $(155, 64)$ kodu, zbog postojanja $(5, 3)$ *trapping set*-ova. Dekoder D_1 razbija većinu kritičnih struktura i omogućava ispravljanje konfiguracija grešaka male tižine, nekorektibilnih pouzdanim *Gal-*



Slika 6.6: Poređenje D_1 i pouzdanog *Gallager A/B* dekodera na LS (155,64) kodu.

lager A/B dekoderom. Međutim, upotreba D_1 dekodera ne dovodi do ispravljanja samo $(5, 3)$ *trapping set*-ova, već i drugih štetnih struktura. Ova činjenica je ilustrovana na slici 6.6, gde su upoređene performanse D_1 i *Gallager A/B* dekodera na kodu LS(155,64) [78]. Kako je to napomenuto u Poglavlju 3, ovaj kod, ima iste strukturne parametre kao *Tanner*-ov kod, ali ne sadrži $(5, 3)$ *trapping set*-ove. Kako je to prikazano u Poglavlju 5, performanse pouzdanog i nepouzdanog *Gallager A/B* dekodera primenjenog na ovaj kod su iste i odgovaraju ispravljanju svih trostrukih grešaka. Čak i na ovom optimizovanom kodu D_1 prevazilazi *Gallager A/B* za čitav red veličine u *error-floor* regionu. Na primer, kada je $\alpha = 0.006$ pouzdani *Gallager A/B* dekoder postiže FER približno 10^{-6}, dok je verovatnoća zaostale greške D_1 dekodera 4×10^{-8}.

Poboljšanje koje postiže D_1 dekoder ilustrovano je i na dužim kodovima, što je prikazano na slici 6.7. Tri 3-levo-regularna koda su razmatrana i to: iRISC(1296,972) [134] dužine 1296, težine vrsta 12 i kodnog količnika 0,75; LS(2388,1793) [78] dužine 2388, težine kolona 12 i kodnog količnika 0,75; Margulis(2640,1320) [87] dužine 2640, težine kolona 12 i kodnog količnika 0,5. Za sva tri koda primećeno je poboljšanje za red veličine u *error-floor* regionu. Najveće poboljšanje postignuto je na *Margulis*-ovom (2640,1320) kodu, gde, kada je $\alpha = 0.02$, D_1 postiže FER od 10^{-6}, dok je za pouzdani *Gallager A/B* dekoder veća od 10^{-4}.



Slika 6.7: FER D_1 i pouzdanog *Gallager A/B* dekodera na dužim 3-levo regularnim kodovima.

6.4 O kompleksnosti dekodera

U ovom odeljku analizirana je kompleksnost MAE dekodera, izražena preko broja 2-ulaznih *Boole*-ovih funkcija korišćenih za implementaciju dekodera po jednom kodnom bitu.

Posmatrana je samo komplekst kombinacione logike, dok je hardverski deo vezan za memorisanje ili druge pomoćne operacije zanemaren.

U svakom varijabilnom čvoru potrebno je implementirati funkciju $F(\{\nu_e^{(\ell)}|e \in \mathcal{E}(v_i)\}, \emptyset)$. To je moguće uraditi koristeći γ-ulazno AND logičko kolo, koje je moguće dekomponovati na $\gamma - 1$ 2-ulazna AND kola. Dodatno, 2 XOR operacije se koriste kada je $\ell = 3 \pmod 4$, dok 2γ XOR kola odgovara slučaju $\ell = 2 \pmod 4$. Tako varijabilni čvorovi doprinose sa $n(3\gamma + 1)$ 2-ulaznih logičkih kola.

Slično, u svakom kontrolnom čvoru ρ-ulazno XOR kolo se koristi kad je $\ell = 0 \pmod 4$. Kada je $\ell = 1 \pmod 4$ ρ-ulazno AND kolo i 2ρ XOR kola se uključuju, dok ostale iteracije zahtevaju ρ $(\rho - 1)$-ulaznih AND kola i 2ρ XOR kola. Svaki od $n\gamma/\rho$ kontrolnih čvorova doprinosi sa $\gamma\rho + 4\gamma - 2\gamma/\rho$ logičkih kola. Kompleksnost MAE dekodera iznosi

$$C_A = \gamma\rho + 7\gamma - 2\gamma/\rho + 1. \tag{6.12}$$

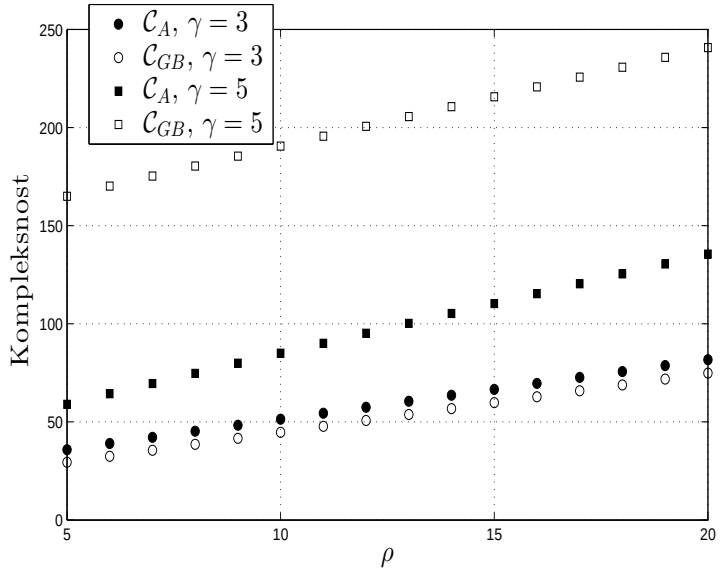
S druge strane kompleksnost *Gallager B* dekodera moguće je izraziti kao

$$C_{GB} = \gamma(\rho + M_k - 1 - 1/\rho) + M_\gamma, \tag{6.13}$$

gde je $k = \gamma - (1 + (-1)^\gamma)/2$, a M_m predstavlja kompleksnost m-ulaznog kola za većinsko odlučivanje (MAJ kola) i može biti izračunata kao (pogledati Poglavlje 7)

$$M_m = \binom{m}{\lceil m/2 \rceil} - 1 + \sum_{i=0}^{\lceil m/2 \rceil - 2} \binom{m-i}{\lceil m/2 \rceil - i}. \tag{6.14}$$

Treba naglasiti da je u $\mathcal{C}_{GB}$ uračunata kompleksnost logike za finalno odlučivanja i računanje sindroma. Kompleksnosti C_{GB} i C_A numerički su izražene na slici 6.8. Primetiti da kada je težina kolona kontrolne matrice 3, MAE dekoder zahteva nešto više logičkih kola od *Gallager B* dekodera i može se primetiti da važi $\rho > 5$ $|C_A - C_{GB}|/C_{GB} \leq 20\%$. Međutim, kompleksnost *Gallager B* dekodera se povećava brzo sa γ i, na primer, kada je $\gamma = 5$ i $\rho \leq 12$ njegova kompleksnost je duplo veća od kompleksnosti MAE dekodera.



Slika 6.8: Poređenje kompleksnosti MAE i *Gallager B* dekodera.

6.5 Zaključak

Mehanizam za razbijanje *trapping set*-ova predložen u ovom poglavlju identifikuje potencijalne štetne strukture i invertuje vrednosti čvorova za koje se sumnja da učestvuju u štetnoj strukturi. Suprotno od standardnog pristupa iterativnih dekodera, gde se odluke o invertovanju

donose na osnvou samo poruka koje šalju susedni čvorovi, u predloženoj strategiji varijabilni čvorovi komuniciraju direktno. Pokazano je da cena koja se plaća uvođenjem dodatnih poruka nije velika i da je kompleksnost predloženog dekodera uporediva sa kompleksnošću *Gallager A/B* dekodera na 3-levo-regularnim kodovima. MAE dekoder može biti konstruisan samo od XOR and AND logičkih kola i značajno je jednostavniji od skorije predloženih PGDBF ili TBBF dekodera. Kompleksnost PGDBF dekodera određena je složenošću generatora slučajnih brojeva potrebnih u svakom varijabilnom čvoru, dok je u TBBF dekoderima potrebno implementirati numerički aparat koji računa dvobitske poruke koje razmenjuju čvorovi grafa.

Pokazano je da hibridni dekoder (kompleksnosti dva puta veće od *Gallager B* dekodera) nadmašuje veći broj dekodera koji donose tvrde ili meke odluke. Predloženi dekoder nije dizajniran za poseban profil *trapping set*-ova i moguće ga je uspešno primeniti na raznim kodovima, kako je to ilustrovano numeričkim rezultatima. Posebno je naglašena univerzalnost MAE dekodera i mogućnost formiranja hibridnih dekodera kombinovanjem MAE dekodera sa drugim iterativnim dekoderima.

Poglavlje 7

Memorije bazirane na LDPC kodovima

Efekti konstantnog skaliranja poluprovodničkih tehnologija – manje dimenzije tranzistora, veća gustina pakovanja struktura, niži nivoi napajanja – utiču značajno na smanjenje pouzdanosti memorijskih uređaja. Istraživanje iz oblasti integrisanih poluprovodničkih struktura značajno je usmereno na rešavanje problema nepouzdanosti [8, 135, 136]. Tradicionalno, pouzdanost memorijskih uređaja obezbeđuje se prostim multipleksiranjem memorijskih ćelija, konceptom koji je predložio *von Neumann* [13]. Posebno je značajna arhitektura nazvana TRC, u kojoj se svaka komponenta multiplicira tri puta [14–16]. Međutim, poznato je da ako je nepouzdanost komponenata memorijskog uređaja velika, velika je i redundansa koju je potrebno dodati [12]. Iako zbog svoje jednostavnosti TRC predstavlja praktično značajno rešenje, sa informacionog stanovišta ona je loša. Kako je *von Neumann*-ovo multipleksiranje u stvari kod sa ponavljanjem, ono je u suprotnosti sa II *Shannon*-ovom teoremom. Upotreba boljih kodova jasno dovodi do unapređenja kako je to pokazao *Spielman* [137], kombinujući *von Neumann*-ove ideje i *Reed-Solomon*-ove kodove. Memorijska arhitektura koja bi se mogla konstruisati na osnovu *Spielman*-ovih zapažanja imala bi redundansu koja logaritamski raste sa povećanjem broja memorijiskih ćelija.

Fundamentalno drugačije ponašanje pokazuju memorije konstruisane pomoću LDPC kodova, kao što je to pokazao *Taylor* [22] u svom pionirskom članku objavljenom šezdesetih godina prošlog veka. *Taylor*-ova memorijska arhitektura pored nepouzdanih memorijskih ćelija koje skladište kodne bite LDPC koda, sadrži i kolo za korekciju grešaka koje se periodično uključuje, dekoduje bite i ponovo ih upisuje u memorijske ćelije. Pritom se smatra da je i logičko kolo implementirano na nepouzdanom hardveru. Izuzetnost *Taylor*-ovog rada ogleda se u činjenici da je pokazao da ovakava memorijska arhitektura implementirana samo od nepouz-

danih komponenti u asimptotskom slučaju može uspešno skladištiti informacije proizvoljno dug vremenski period, a da pri tom redundansa memorije ostaje konstantna. Istu memorijsku arhitekturu analizirao je *Kuznetsov* [23], koji je poboljšao uslove koji obezbeđuju uspešno skladištenje informacija, pa se često u literaturi ovakav teoretski koncept naziva *Taylor-Kuznetsov*-a memorija.

Revolucionarne tvrdnje *Taylor*-a i *Kuznetsov*-a bile su ispred tehnoloških dostignuća svog vremena, pa su dugo bila zapostavljena i istraživanjiva koja obuhvataju informacioni aspekt memorija. Tako da osim značajne studije *Pippenger*-a [21] ozbiljnijih pokušaja analize pouzdanih kodovanih sitema nije bilo sve do istraživanja Vasića i *Chilappagari*-ja [12]. Pomenuti autori uočili su da se ciklus ažuriranja memorijskih ćelija može ekvivalentirati jednom iteracijom *Gallager B* algoritma dekodovanja LDPC kodova, što je usmerilo istraživanje ka analizi dekodera opisanih grafovima. Isti autori su u [24, 138] predložili memorijsku arhitekturu baziranu na *bit-flipping* dekoderu, pri tom dokazujući da i ona ima sposobnost pouzdanog čuvanja informacija u asimptotskom slučaju. Dodatno, *Dupraz* [139] je analizirala koliki nivo nepozdanosti ovakva memorija može tolerisati, kada se njena nepouzdanost opisuje *von Neumann*-ovim modelom otkaza. S druge strane, Ivković [140] je pokazao da se performanse memorijskih arhitektura mogu unaprediti dodavanjem kola za računanje sindroma, koja moraju biti savršeno pouzdana. *Varshney* [25] je koristeći *density evolution* analizu pokazao da teorijski značajne performanse postiže i arhitektura koja uključuje *Gallager A* dekoder. Značajno je pomenuti istraživanjnja *Khajeh*-a [141] koji je posmatrao *buffer* memorije i predložio združeni opis otkaza logičkih kola i nepouzdanosti komunikacionog kanala.

Do sada u literaturi nije bilo analitičkog pristupa koji bi uključio uticaj korelisanih otkaza logičkih kola na sposobnost kodovane memorije da skladišti informacije. U ovom poglavlju posmatrana je memorija bazirana na *bit-flipping* dekoderu, čiji su otkazi posledica smanjenja napona napajanja i mogu se opisati GOS modelom. Na osnovu garantovane korektivne sposobnosti nepouzdanog *bit-flipping* dekodera date u Poglavlju 4, procenjeno je koliki deo svih komponenti može da otkaže između dva ciklusa ažuriranja, a da memorija i dalje uspeva da radi pouzdano. Dodatno, pokazano je da u asimptotskom slučaju memorija uspeva da sačuva sve informacije, što je prvi takav rezultat na nekom drugom modelu osim *von Neumann*-ovog. Analizički rezultati ilustrovani su i numerički. Prezentovani rezultati objavljeni su radovima [142, 143].

Ostatak poglavlja organizovan je na sledeći način. U Odeljku 7.1 opisana je ideja *Tay-*

lor-ove kodovane memorijske arhitekture, dok je generalniji opis kodovanih memorija dat u Odeljku 7.2. Odeljak 7.3 posvećen je analizi *bit-flipping* memorije u prisustvu korelisanih otkaza u logičkim kolima. Potencijalna primena prezentovanih rezultata na neke praktično značajne tehnologije izrade memorija napomenuta je u Odeljku 7.4, dok su zaključne napomene izložene u Odeljku 7.5. Pojedina matematička izvođenja izmeštena su iz glavnog teksta i priložena u dodatku.

7.1 Kodovana memorija i *Taylor-Kuznetsov* koncept

Memorija je elektronski uređaj koji skladišti informacije da bi se mogle koristiti kasnije. Ona ima zadatak da obezbedi skladištenje informacija što je moguće duži vremenski period, kao i njihovu upotrebu u proizvoljno izabranom vremenskom trenutku. Informacija se čuva u memorijskim ćelijama koje imaju mogućnost skladištenja binarnih informacija. Pritom se smatra da su ćelije nepouzdane i da se informacije vremenom gube. Nepouzdanost memorijskih ćelija posledica je različitih varijacija u procesu proizvodnje uređaja, smanjenja napona napajanja, kao i poremećenog praga uključivanja tranzistora i može se manifestovati kao slučajno invertovanje sadržaja ćelije. Greške utiču samo na trenutnu vrednost ćelije i ne prouzrokuju oštećenja ćelije, i često se u literaturi nazivaju mekim greškama (eng. *soft errors*).

S druge strane, usled deformacije ćelijske strukture poznato je da ćelija može otkazati i trajno (permanentno), kada govorimo o tvrdim greškama (eng. *hard errors*) [46, 144]. Tada ćelija ne dozvoljava upis nove vrednosti, što se u engledskoj literaturi često naziva *stuck-at* defektom. Memorijski koncept koji su razvili *Taylor* [22] i *Kuznetsov* [23] zahteva periodično ažuriranje vrednosti memorijskih lokacija, pa nije direktno primenljiv za ispravljanje tvrdih grešaka. Umesto njihovog principa rasprostranjena je upotreba particionisanih linearnih blok kodova [145, 146], sa zadatkom da se informacije o tvrdim greškama inkorporiraju u proces kodovanja, kada se pogodnim izborom kodnih reči potencijalni otkazi memorijskih ćelija maskiraju. Drugačiji pristup problemu tvrdih grešaka zahteva iplementaciju pokazivača grešaka (eng. *error-correction pointers*) kojima se specificira adresa ćelije koja je oštećena, da bi se ona kasnije zamenila ispravnom ćelijom, kako je to predloženo u [46, 144]. U ovom poglavlju će se uglavnom analizirati problem mekih grešaka, ali će biti skrenuta pažnja da predložena memorijska arhitektura omogućava čuvanje informacija i kada su otkazi permanentni.

Da bi ilustrovali potrebu za kodovanim memorijskim uređajima, možemo posmatrati izolo-

vanu memorijsku ćeliju koja čuva jedan bit informacija. Neka je u trenutku $t = 0$ u ćeliju upisana binarna vrednost. Verovatnoća da sadržaj ćelije bude invertovan u toku trajanja jediničnog vremenskog intervala τ jednaka je p_m. Drugim rečima u informacionom smislu sadržaj memorijske ćelije se periodično, u vremenskim intervalima $\tau - \delta, 2\tau - \delta, \ldots, L\tau - \delta, \ldots$, gde je δ beskonačno mali vremenski interval, propušta kroz binarni simetrični kanal i skladišti ponovo u ćeliji. Tada je verovatnoća na se kraju intervala posmatranja $L\tau$ u ćeliju upiše pogrešna vrednost, $P_f(L\tau)$, jednaka [23]

$$P_f(L\tau) = \frac{1 - (1 - 2p_m)^{L\tau}}{2}. \tag{7.1}$$

Može se primetiti da $\lim_{L\to\infty} P_f(L\tau) = 0,5$, pa se nakon dovoljno dugog vremena informacija gubi. Da bi se skladištena informacija zaštitila, k memorijskih ćelija moguće je zaštititi nekim linearnim blok kodom, što će uneti određenu redundansu u uređaj i zahtevati skladištenje k/R bita, gde je R kodni količnik primenjenog koda. Pritom za kodovani uređaj, na osnovu druge *Shannon*-ove teoreme, sleduje

$$P_f(\tau) \leq e^{-\frac{k}{R}E(R)}, \tag{7.2}$$

gde je $E(R)$ neka pozitivna vrednost za svako $R < 1 - H(p_m)$, gde je $1 - H(p_m)$ označen kapacitet binarnog simetričnog kanala verovatnoće greške p_m. Iako je linearni blok kod ograničio verovatnoću greške u nekom intervalu τ, naredni interval τ doneće nove greške u memorijskim lokacijama. Zbog toga je potrebno nakon unosa grešaka, sadržaj memorijskih ćelija periodično ažurirati, tj. u trenucima $\ell\tau$, $\ell > 0$, kodnu reč pročitati iz memorije, dekodovati i ponovo upisati na iste pozicije. Tada se na osnovu gornje granice unije korelisanih događaja (eng. *union bound*) može tvrditi sledeće [22]

$$P_f(L\tau) < Le^{-\frac{k}{R}E(R)}. \tag{7.3}$$

Kako se gornaja granica verovatnoće greške najviše linerano povećava sa brojem ciklusa ažuriranja, moguće je pronaći dovoljno dug kod koji će garantovati proizvoljno malu verovatnoću pogrešnog čitanja informacije iz memorije u svakom trenutku. Za takve memorije kaže se da su *stabilne*. Primetiti da je prethodna diskusija smatrala da proces dekodovanja savršeno pouzdan. Da bi olakšao analizu i dizajniranje stabilnih memorija *Taylor* je definisao pojmove kompleksnosti, redundanse i stabilnosti, koji su navedeni u nastavku.

Definicija 7.1. *Informacioni kapacitet memorije predstavlja broj informacionih bita koji se skladište u memoriji.*

Definicija 7.2. *Kompleksnost memorije M_k, informacionog kapaciteta k, definiše se kao ukupan broj memorijskih ćelija i 2-ulaznih Boole-ovih funkcija korišćenih u memorijskoj arhitekturi.*

Definicija 7.3. *Redundansa memorije $\mathcal{R}$ je odnos kompleksnosti memorije i memorije koja ne sadrži redundansu, a ima isti informacioni kapacitet.*

Definicija 7.4. *Otkaz memorije u trenutku t proglašava se samo onda kada sadržaj memorije nije moguće uspešno dekodovai pouzdanim dekoderom.*

Primetiti da i koncept kodovanih memorija zahteva "zlatna" logička kola implementirana u vidu pouzdanog dekodera, koji se uključuje onda kada je potrebno koristiti sadržaj skladišten u memoriji. *Taylor* je takođe uveo pojam *klase konvergencije* koja sadrži potencijalne sekvence, koje mogu da se pojave prilikom čitanja informacija iz memorije, a moguće ih je uspešno dekodovati istim dekoderom korišćenim za ažuriranje memorijskih ćelija. S druge strane, *Varshney* [25] je predložio da se za proces čitanja iz memorije koristi dekoder koji radi po principu maksimalne verodostojnosti (ML dekoder).

Definicija 7.5. *Memorija M_k je stabilna ako je zadovoljeno sledeće:*

i) Kompleksnost memorije M_k je ograničena sa θK, gde je θ fiksan parametar.

ii) Za svaki vremenski trenutak $t > 0$, i $\delta > 0$, verovatnoća otkaza memorije u trenutku t zadovoljava $P_f(t) < \delta$.

Prvi navedeni uslov govori o tome da stabilna memorija mora da ima konačnu redundansu, dok drugi uslov govori o asimptotskim osobinama memorije. Interesantno je primetiti da dekoder uključen u stabilnu memoriji mora imati kompleksnost $O(k)$, dok je poznato da je kompleksnost praktično značajnih dekodera blok kodova $O(k \log k)$. Međutim, iterativni dekoderi LDPC kodova prostornu kompleksnost zamenjuju vremenskom, tako da je dovoljno fizički implementirati samo logička kola dovoljna za jednu iteraciju dekodovanja. Tada član $\log k$ nestaje, a iterativni proces dekodovanja postaje proces periodičnog ažuriranja memorijskih ćelija. Treba naglasiti da su LDPC kodovi jedini blok kodovi koji zadovoljavaju prvi uslov stabilnosti memorije. Oni u načelu zadovoljavaju i drugi uslov jer dostižu *Shannon*-ov kapacitet.

Značaj *Taylor*-ov rada ogleda se u činjnici da je uspeo da dokaže da memorija sastavljena samo od nepouzdanih komponenti može da bude stabilna. Arhitekura memorije koju je

predložio *Taylor*, a kasnije usavršio *Kuznetsov* data je na slici 7.2.a. Informacija dužine k bita kodovana je regularnim binarnim LDPC kodom dužine n u oznaci $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Vektor sindroma označen je sa $\mathbf{c} = (c_1, c_2, \ldots, c_m)$, dok je skup provera parnosti u kojim učestvuje bit x_i označen sa $\{c_i^{(1)}, c_i^{(2)}, \ldots, c_i^{(\gamma)}\}$, gde je γ težina kolona kontrolne matrice LDPC koda. Nakon kodovanja, koje se izvršava pouzdanim logičkim kolima, u memoriji se skladišti γ kopija svakog bita, u oznaci $\{x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(\gamma)}\}$, organizovanih u γ registara tako da jedan registar sadrži jednu kopiju kodne reči. Inicijalno sve kopije bita imaju istu vrednost. Registri su napravljeni od nepouzdanih ćelija, a njihove vrednosti se ažuriraju na osnovu jedne kombinacije od $\gamma - 1$ provera parnosti. Ciklusi ažuriranja dati su u nastavku.

* Izračunati provere parnosti za svaku kopiju kodnog bita, pritom isključiti jednu od γ provera parnosti.

* Invertovati vrednost svake kopije, ako ona učestvuje u više od polovine nezadovoljenih provera parnosti.

* Periodično ponavljati prethodna dva koraka.

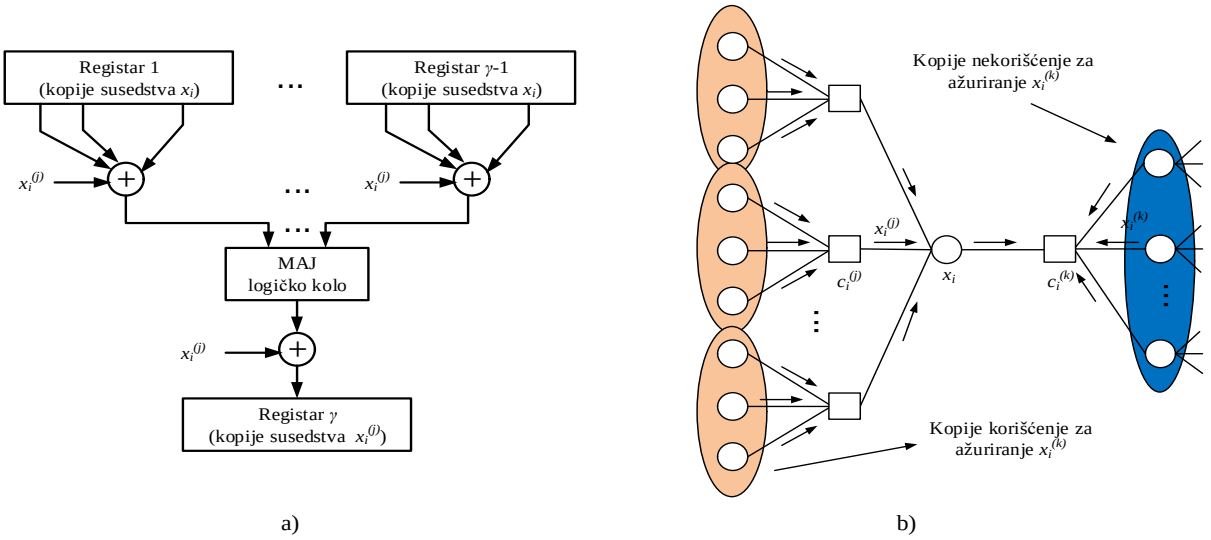
Prilikom ažuriranja kopije $x_i^{(j)}$ ne koriste se kopije $x_l^{(k)}$ koje su povezane na proveru parnosti $c_i^{(k)}$, koja je isključena iz ažuriranja ovog bita. Pritom se prilikom ažuriranja neke druge kopije bita x_i isključuje uvek različita provera parnosti. U vreme pionirskog *Taylor*-ovog rada, opis LDPC kodova putem *Tanner*-ovog grafa nije bio poznat. Vasić i *Chilappagari* [12] su 2006. godine uočili da je proces ažuriranja memorijskih registara ekvivalentan jednoj iteraciji *Gallager B* algoritma (slika 7.2.b), što je omogućilo primenu znanja vezanog za LDPC kodove akumuliranog tokom prethodnih decenija.

Treba naglasiti da je *Kuznetsov* analizirao istu memorijsku arhitekturu i poboljšao uslove koji dovode do stabilnosti memorije. Neka je verovatnoća otkaza memorijskih ćelija p_m, dok XOR i MAJ logička kola otkazuju sa verovatnoćama $p_\oplus$ i p_γ, respektivno. Tada verovatnoća otkaza *Taylor*-ove memorije nakon vremena $L\tau$ iznosi [23]

$$P_f(L\tau) \leq A(p_m, p_\oplus, p_\gamma) L n^{-\beta(p_m, p_\oplus, p_\gamma)} \Big(\frac{1}{2\alpha} + \ln n \Big), \tag{7.4}$$

gde se pokazuje da postoje $A, \beta > 0$, dok je

$$\alpha = \frac{1}{2\ln(\gamma - 1)(\rho - 1)}, \tag{7.5}$$

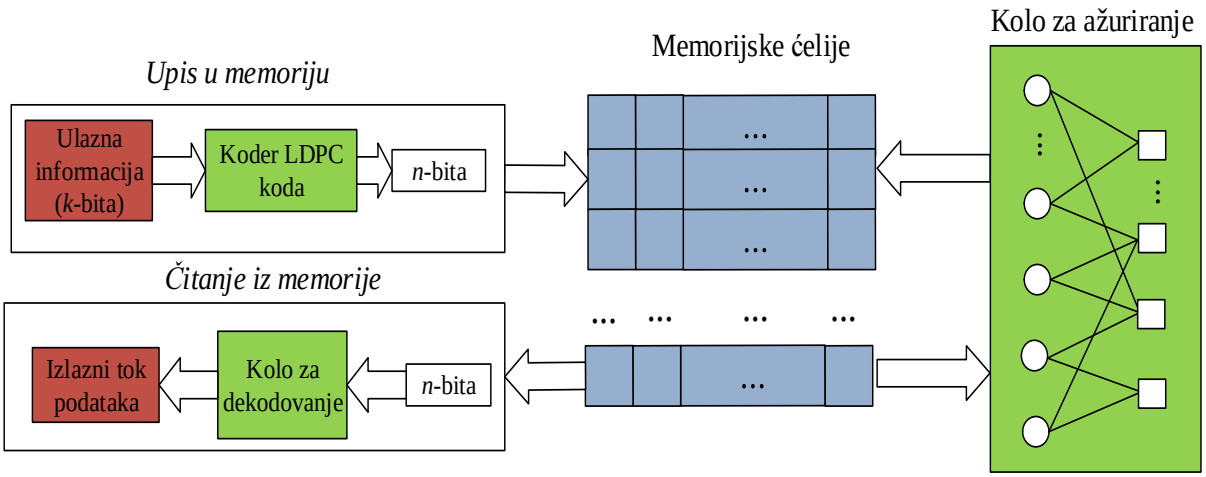


Slika 7.1: *Taylor-Kuznetsov*-a arhitektura: a) blok dijagram; b) opis grafovskom strukturom .

gde je sa ρ označena težina vrsta kontrolne matrice koda. Iako je *Taylor* došao do sličnog rezultata vrednost parametra β koju je odredio *Kuznetsov* je veća, pa samim tim i robustnost memorije na otkaze komponenti je veća.

7.2 Okvir istraživanja memorija baziranih na LDPC kodovima

U ovom odeljku prikazan je okvir za istraživanje memorija na bazi LDPC kodova. Predstavljene su različite memorijske arhitekture za koje je poznato da su stabilne u *Taylor*-ovom smislu, kada se otkazi memorijskih komponenti opisuju *von Neumann*-ovim modelom. Generalizovana arhitektura kodovanih memorija data je na slici 7.2. Informacija se koduje LDPC kodom tako da se na izlazu iz kodera pojavljuje kodna reč $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Kodna reč se skladišti u nepouzdanim memorijskim ćelijama, koje se periodično ažuriraju na osnovu kola za korekciju grešaka. Pri tom se razlikuju dva tipa memorija: (i) memorije na bazi BF dekodovanja gde se čuva samo jedna kopija kodnog bita, i (ii) memorije na osnovu *message-passing* dekodera kada je potrebno skladištiti γ kopija svakog bita, gde je sa γ označena težina kolona kontrolne matrice koda. Nepouzdanost memorijskih ćelija opisuje se n-dimenzionim binarnim nizom $\mathbf{Y}$ definisanim nad $\{0,1\}^n$ sa nezavisnim elementima Y_j takvim da je $\Pr\{Y_j = 1\} = p_m$, $1 \leq j \leq n$. Neka se ažuriranje memorijskih lokacija obavlja u trenucima $t = \ell\tau$, $\ell > 0$ i neka je $\mathbf{r}^{(i)}(t) = (r_1^{(i)}, r_2^{(i)}, \ldots, r_n^{(i)})$ vektor vrednosti koje odgovaraju i-toj kopiji kodne reči, skladištenim u memorijskim lokacijama u trenutku t. Ako se kolo za korekciju grešaka bazira



Slika 7.2: Generalizovana blok šema memorija baziranih na LDPC kodovima.

na BF algoritmu $i = 1$, dok za *message-passing* dekodere može uzimati vrednosti iz opsega $1 \leq i \leq \gamma$. Vrednosti pročitane iz memorije u trenutku pre ažuriranja $\ell\tau - \delta$ su

$$\mathbf{r}^{(i)}(\ell\tau - \delta) = \mathbf{r}^{(i)}((\ell - 1)\tau + \delta) \oplus \mathbf{y}^{(\ell)}, \tag{7.6}$$

gde $\mathbf{y}^{(\ell)}$, označava ℓ-tu partikularnu realizaciju slučajne promenljive $\mathbf{Y}$. Pri tome važi $\mathbf{r}^{(i)}(\tau - \delta) = \mathbf{x} \oplus \mathbf{y}^{(\ell)}, \forall i$. Kombinaciona logika kola za korekciju organizovana je na bazi *Tanner*-ovog grafa pri čemu je poznato da stabilnost memorije garantuje upotreba *bit-flipping*, *Gallager A* ili *Gallager B* dekodera. Poruka koju kontrolni čvor c šalje preko grane e, ν_e, za sva tri algoritma računa se na isti način, pa imamo

$$\nu_e = \bigoplus_{e' \in E(c) \setminus \{e'\}} \mu_{e'}, \tag{7.7}$$

gde je $\mathcal{E}(c)$ skup grana koje povezuju kontrolni čvor c sa susednim varijabilnim čvorovima, dok μ_e predstavlja kopiju bita $v \in \mathcal{N}(c)$ pročitanu iz memorije. Pri tom za BF dekoder imamo da je $\mu_e = r_v^{(1)}(\ell\tau - \delta)$, dok za *message-passing* dekodere $\mu_e = r_v^{(e)}(\ell\tau - \delta)$. U svakom kontrolnom čvoru potrebno je implementirati ρ XOR kola sa $\rho - 1$ ulaza.

Vrednosti koje se upisuju u memoriju u trenutku $\ell\tau + \delta$, ako kolo za korekciju odgovara BF dekoderu, iznose

$$r_v^{(1)}(\ell\tau + \delta) = \begin{cases} s, \text{ ako je } |\{e \in \mathcal{E}(v) : \nu_e = s\}| \geq \lceil \frac{\gamma}{2} \rceil, \\ r_v^{(1)}(\ell\tau - \delta_0), \quad \text{inače}, \end{cases} \tag{7.8}$$

gde je $\mathcal{E}(c)$ skup susednih varijabilnih čvorova čvoru c, $s \in \{0, 1\}$, dok $\lceil \gamma/2 \rceil$ označava najmanji prirodan broj veći od $\gamma/2$. Ovakav tip memorije zahteva implementaciju jednog γ-ulaznog MAJ logičkog kola. Ako se memorija bazira na *message-passing* dekoderu u memoriju

se upisuju sledeće vrednosti

$$r_v^{(e)}(\ell\tau+\delta_0)=\begin{cases} s, & \text{ako je } |\{e'\in\{\mathcal{E}(v)\setminus e'\}:\nu_{e'}=s\}|\geq b,\\ r_v^{(e)}(\ell\tau-\delta), & \text{inače,}\end{cases} \tag{7.9}$$

gde vrednost praga b određuje o kom *message-passing* dekoderu je reč. Ako je $b=\lceil\gamma/2\rceil$, govorimo o memoriji baziranoj na *Gallager B* dekoderu, dok $b=\gamma-1$ odgovara *Gallager A* dekoderu. Memorija sa *Gallager B* dekoderom zahteva implementaciju $(\gamma-1)$-ulaznog MAJ logičkog kola za ažuriranje svake memorijske ćelije, dok se, ako se koristi *Gallager A* dekoder, implementiraju $(\gamma-1)$-ulazna komparatorska kola.

Poznato je da se $\rho-1$-ulazno XOR kolo može dekomponovati na $(\rho-2)$ 2-ulaznih XOR logičkih kola. Slično se i $(\gamma-1)$-ulazno komparatorsko kolo može implementirati kao konkatenacija $\gamma-2$ 2-ulaznih komparatorskih kola. Tada se redundansa memorije koja sadrži *Gallager A* dekoder, $\mathcal{R}_{GA}$, može biti ograničena kao [25]

$$\mathcal{R}_{GA}\leq(\gamma\rho-1)/((1-\gamma/\rho)). \tag{7.10}$$

Da bi pronašli redundansu ostale dve memorijske arhitekture potrebno je prvo odrediti kompleksnost višeulaznog MAJ logičkog kola, što je dato u sledećoj lemi.

Lema 7.1. *Kompleksnost γ-ulaznog MAJ logičkog kola, $\gamma\geq 3$ zadovoljava*

$$D_\gamma\leq\binom{\gamma}{\lceil\gamma/2\rceil}-1+\sum_{i=0}^{\lceil\gamma/2\rceil-2}\binom{\gamma-i}{\lceil\gamma/2\rceil-i}. \tag{7.11}$$

Dokaz: Boole-ova funkcija koja obavlja većinsko odlučivanja između γ ulaza može biti dekomponovana na dva dela: prvi deo koji ispituje svaku kombinaciju od $\lceil\gamma/2\rceil$ ulaza i drugi deo koji skuplja informacije od prvog dela. Prvi deo zahteva implementaciju $\binom{\gamma}{\lceil\gamma/2\rceil}$ $\lceil\gamma/2\rceil$-ulaznih AND kola, dok $\binom{\gamma}{\lceil\gamma/2\rceil}$-ulazno OR kolo se koristi u drugom delu. Poznato je da se m-ulazno AND kolo može predstaviti kao veza $m-1$ 2-ulaznh AND kola. S druge strane, ako bi izvršili takvu dekompoziciju AND kola, neka 2-ulazna AND kola bi se pojavila više puta u implementaciji. Da bi se izbegla takva situacija i odredio minimalni broj potrebnih logičkih kola, izvršena je paralelna dekompozicija. Neka je $f_{\lceil\gamma/2\rceil}(z_1,z_2,\ldots,z_{\lceil\gamma/2\rceil})$ funkcija koja predstavlja $\lceil\gamma/2\rceil$-ulazno AND kolo, gde su ulazni argumenti $z_1,z_2,\ldots,z_{\lceil\gamma/2\rceil}$ izabrani iz skupa od γ ulaza u MAJ kolo. Ova funkcija se može predstaviti kao

$$\begin{aligned} &f_{\lceil\gamma/2\rceil}(z_1,z_2,\ldots,z_{\lceil\gamma/2\rceil})\\ &=f_{\lceil\gamma/2\rceil-1}(z_1,z_2,\ldots,x_{\lceil\gamma/2\rceil-1})z_{\lceil\gamma/2\rceil}, \end{aligned} \tag{7.12}$$

što dovodi do $\binom{\gamma}{\lceil\gamma/2\rceil}$ različitih 2-ulaznih AND kola. Ako se prethodno navedena dekompozicija nastavi, dobija se $f_{\lceil\gamma/2\rceil-1}(z_1, z_2, \ldots, z_{\lceil\gamma/2\rceil-2})z_{\lceil\gamma/2\rceil-2}$, gde je kardinalni broj skupa mogućih ulaza redukovan na $\gamma - 1$. Tako se dobija dodatnih $\binom{\gamma-1}{\lceil\gamma/2\rceil-1}$ različitih 2-ulaznih AND logičkih kola. Dalja iterativna dekompozicija dovodi do vrednosti date u nejednakosti (7.11). ■

Sada se mogu predstaviti redundanse memorija baziranih na *Gallager B* i BF dekoderima, $\mathcal{R}_{GB}$ i $\mathcal{R}_{BF}$, respektivno, kao

$$\begin{aligned}\mathcal{R}_{GB} &\leq \gamma(1 + D_{\gamma-1} + \gamma(\rho - 2))/(Rn) \\ &\leq \gamma(1 + D_{\gamma-1} + \gamma(\rho - 2))/((1 - \gamma/\rho)),\end{aligned} \tag{7.13}$$

$$\mathcal{R}_{BF} \leq (1 + D_{\gamma} + \gamma(\rho - 2))/((1 - \gamma/\rho)). \tag{7.14}$$

Redundanse različitih memorijskih arhitektura numerički su izražene na slici 7.3, za različite vrednosti parametra ρ i fiksno γ. Može se primetiti da za zadati algoritam dekodovanja postoje optimalni parametri koji dovode do memorije sa najmanjom redundansom, pri čemu izbor optimalnog koda ne zavisi od dekodera koji se primenjuje. Primećuje se da je redundansa memorije sa *Gallager B* dekoderom grubo γ puta veća od redundansi preostale dve memorijske arhitekture.

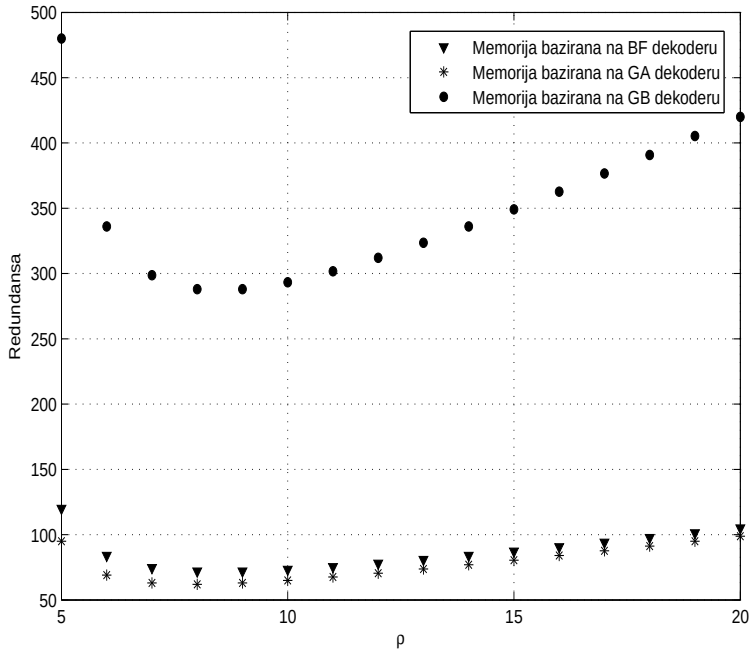
Postavlja se pitanje koja je najmanje kompleksna memorija, koja ispunjava *Taylor*-ove uslove stabilnosti. *Taylor* je formalizovao ovo pitanje i definisao pojam *kapaciteta memorije* (eng. *storage capacity*).

Definicija 7.6. *Kapacitet memorije, $\mathcal{C}$, predstavlja brojnu vrednost takvu da postoje stabilne memorije za sve redundanse veće od* $1/\mathcal{C}$.

Nažalost, *Taylor* nije numerički izrazio kapacitet, niti je do sada bilo ozbiljnijih pokušaja da se on odredi. *Varshney* [25] koji je dokazao stabilnost memorije koja uklkučuje *Gallager A* dekoder pokazao je da za $\gamma = 3$ minimum funkcije date izrazom (7.10) 34, pa $\mathcal{C} > 1/34$ predstavlja najužu poznatu granicu.

7.3 Performanse memorije sa BF dekoderom

Da memorijska arhitektura koja koristi BF dekoder može biti stabilna prvi su dokazali *Chilappagari* i Vasić [24, 138]. Metodologija dokaza koju su usvojili autori, oslanja se na tzv.



Slika 7.3: Radundanse različitih stabilnih memorija ($\gamma = 4$).

model protivnika (eng. *adversarial model*) i omogućava da se dodatno proceni i broj (frakcija) otkaza komponenti koje memorijska arhitektura može tolerisati. S druge strane, stabilnost memorije bazirane na *Gallager A* dekoderu sledi iz *density evolution* asimptotske analize koju nije moguće koristiti za analizu kodova konačne dužine.

Neka je α_m frakcija memorijskih ćelija koje mogu otkazati u vremenskom intervalu između dva ciklusa ažuriranja $((\ell-1)\tau$,$\ell\tau)$, $\ell > 0$. Prema modelu protivnika sa povećanjem broja memorijskih ćelija, broj dozvoljenih otkaza se takođe povećava. Slično, neka su frakcije XOR i MAJ kola koja mogu otkazati u toku jednog ciklusa ažuriranja označene sa $\alpha_\oplus$ i α_γ, respektivno. Pri tome se priroda otkaza ne razmatra, već se samo ograničava njihov broj. Pod navedenim uslovima *Chilappagari* i Vasić su uspeli da pokažu da memorijska arhitektura može tolerisati frakciju otkaza svih komponenti, što je predstavljeno u sledećoj teoremi.

Teorema 7.1. *Neka je dat* $(\gamma, \rho, \alpha, (3/4+\epsilon)\gamma)$ *ekspander kod. Memorijska arhitektura bazirana na BF dekoderu može tolerisati fiksnu frakciju otkaza komponenti ako je*

$$\alpha_m + \gamma\alpha_\oplus + \alpha_\gamma < 2\epsilon\alpha(1+4\epsilon). \tag{7.15}$$

Dokaz: Pogledati [24]. ■

Ako se sa $\alpha_{total} = \alpha_m + \gamma\alpha_\oplus + \alpha_\gamma$ označi frakcija komponenti koje se mogu tolerisati, tada

se može pronaći i broj otkaza koji neće dovesti do otkaza memorije.

Lema 7.2. *Neka je dat kod iz ($\gamma \geq 8$, $\rho \geq \gamma$)-regularnog ansambla, čiji Tanner-ov graf ima girth $g = 2g_0$. Tada, memorijska arhitektura može tolerisati $\alpha_{total}n$ otkaza komponenti ako je*

$$\alpha_{total}n < 3n_0(\gamma/4, g_0)/8, \tag{7.16}$$

gde je

$$\begin{aligned} n_0(\gamma/4, g_0) &= n_0(\gamma/4, 2j+1) = 1 + \frac{\gamma}{4}\sum_{i=0}^{j-1}\left(\frac{\gamma}{4}\right)^i, \ g_0 \ \textit{neparno}, \\ n_0(\gamma/4, g_0) &= n_0(\gamma/4, 2j) = 2\sum_{i=0}^{j-1}\left(\frac{\gamma}{4}\right)^i, \ g_0 \ \textit{parno}. \end{aligned} \tag{7.17}$$

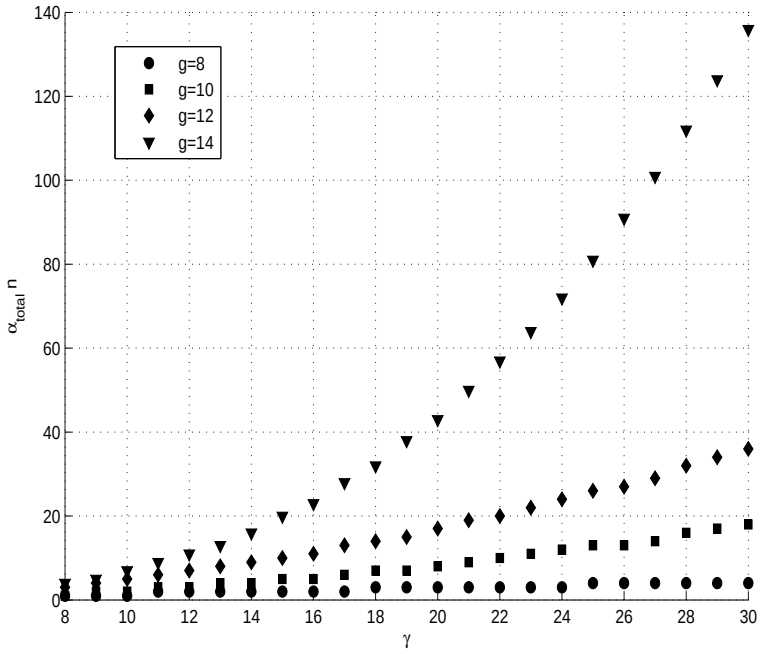
Dokaz: Na osnovu Teoreme 6.1 sledi da kada je ekspanzija koda veća od $7/8\gamma$ ($\epsilon > 1/8$) memorija može tolerisati $3\alpha/8$ otkaza komponenti. S druge strane na osnovu Teoreme 4.3 dobijamo da ekspanziju veću od $7/8\gamma$ ostvaruje najmanje $n_0(\gamma/4, g_0)$ varijabilnih čvorova. ■

Numerički određene vrednosti $\alpha_{total}n$ ilustrovane su na slici 7.4. Primećuje se da za male vrednosti *girth*-a (na primer ako je $g = 8$) broj otkaza koji se mogu tolerisati je nizak, čak i za velike vrednosti γ. S druge strane, povećanje g dovodi do brzog rasta broja otkaza koji se toleriše. Na primer, ako je $\gamma = 15$ za $g = 12$ memorijska arhitektura može tolerisati 12 otkaza, dok za $g = 14$ broj otkaza može rasti do 20.

Već je napomenuto da model protivnika ne uzima u obzir prirodu nastanka otkaza. Postavlja se pitanje da li se gornja granica frakcije otkaza koji se tolerišu može unaprediti, ako se smatra da su otkazi logičkih kola korelisani. Slično kao što je to posmatrano u prethodnim poglavljima, i ovde će biti analiziran uticaj smanjenja napajanja što se opisuje GOS modelom, definisanim u Poglavlju 2. Zbog kompletnosti izlaganja ukratko će biti ponovljen opis ovog modela otkaza.

Neka je $z(\ell\tau)$ ispravan izlaz logičkog kola na kraju $\ell\tau$ ciklusa osvežavanja. Usled nepouzdanosti logičkog kola stvarni izlaz logičkog kola je $z(\ell\tau) \oplus \xi(\ell\tau)$, gde je $\xi(\ell\tau) \in \{0, 1\}$ vrednost greške u trenutku $\ell\tau$. U GOS modelu grešaka, ako ne dolazi do promene vrednosti izlaza kola izlaz je uvek ispravan $\Pr\{\xi(\ell\tau) = 1 | z(\ell\tau) = z((\ell-1)\tau)\} = 0$. S druge strane, logičko kolo nije u stanju da ispravno promeni vrednost sa verovatnoćom $\Pr\{\xi(\ell\tau) = 1 | z(\ell\tau) \neq z((\ell-1)\tau)\} = p_g$, $p_g > 0$, $g \in \{\oplus, \gamma\}$, gde γ označava γ-ulazno MAJ kolo.

U ovom poglavlju će prvo biti određena frakcija otkaza koju je moguće tolerisati, a zatim će se koristeći *Chernoff*-ovu granicu dokazati stabilnost memorije u prisustvu korelisanih otkaza.



Slika 7.4: Broj otkaza koje toleriše memorijska arhitektura.

Pritom se u analizi koja sledi smatra najgori mogući slučaj, u kome otkazi logičkih kola uvek dovode do povećanja nepouzdanosti memorijskih ćelija.

Uticaj otkaza logičkih kola u toku inicijalnog ciklusa ažuriranja detaljno je analiziran u prethodnim poglavljima. Predloženo je praktično rešenje koje omogućava da se prvi ciklus obavi bez otkaza logičkih kola, što je podrazumevano u analizi koja sledi. Prvo je istražena frakcija otkaza memorijskih ćelija α_m koja se može tolerisati, ako su sva logička kola koja se koriste u memoriji nepozdana i otkazuju prema GOS modelu.

Lema 7.3. *Memorijska arhitektura bazirana na LDPC kodu iz (γ, ρ)-regularnog ansambla LDPC, u kome nema ciklusa dužine četiri, može da toleriše frakciju od α_m otkaza između dva ciklusa ažuriranja ažuriranja ako je*

$$\alpha_m \leq \left\lfloor \gamma/6 \right\rfloor / n. \tag{7.18}$$

Dokaz: Ako se za ažuriranje koristi savršeno pouzdano kolo za korekciju grešaka, svaki LDPC kod kod koga nema ciklusa dužine četiri, u toku jednog ciklusa ažuriranja može ispraviti $\lfloor \gamma/2 \rfloor$ pogrešnih vrednosti skladištenih u memorijskim ćelijama. Drugim rečima, svaki kodni bit će biti dekodovan korektno ako broj pogrešnih procena na ulazu u MAJ logičko kolo nije veći od $\lfloor \gamma/2 \rfloor$. U arhitekturi u kojoj je kolo inherentno nepouzdano, pogrešna procena bita se

može pojaviti kao posledica otkaza odgovarajućeg XOR kola. Drugim rečima N_m grešaka u memoriji mogu dovesti do N_m pogrešnih procena. Slično, $N_\oplus$ otkaza XOR kola dovode do $N_\oplus$ pogrešnih procena. Tada se bit neće dekodovati pogrešno ako je

$$N_m + N_\oplus \leq \lfloor \gamma/2 \rfloor. \tag{7.19}$$

Ako je ukupan broj grešaka u vremenskom intervalu τ ograničen prethodnom nejednakošću, nakon svakog ciklusa ažuriranja samo ispravne vrednosti bita će biti upisaivane u memorijske ćelije, što za posledicu ima da će MAJ logička kola uvek raditi ispravno.

Neka je sa $\mathcal{F}_\ell$ označen skup memorijskih ćelija u kojima su skladištene pogrešne vrednosti u vremenskom intervalu između $(\ell-1)$-og i ℓ-tog ciklusa ažuriranja. Sledi da je $|\mathcal{F}_\ell| = N_m \leq \alpha_m n$, za svako $\ell > 0$. Jasno je da maksimalan broj otkza XOR logičkih kola odgovara slučaju kada je $\mathcal{F}_{\ell-1} \cap \mathcal{F}_\ell = \emptyset$. Tada je maksimalan broj otkaza XOR kola upotrebljenih za dekodovanje nekog bita ograničen na $N_\oplus \leq 2\alpha_m n$. ■

Primetiti da se α_m smanjuje kao posledica povećanja dužine koda n, pa broj otkaza koji se tolerišu ne može preći $\lfloor \gamma/6 \rfloor$. To znači da se proizvoljno mala verovatnoća otkaza memorije može postići samo ako γ teži ka beskonačnosti. Glavni razlog za ovakvo ponašanje ogleda se u činjenici da, u slučaju GOS modela, otkazi MAJ logičkih kola mogu poništiti sve odluke koje su dovele do korekcije grešaka. Jedini način da se poveća tolerancija memorijske arhitekture je da se dozvoli određenom broju MAJ logičkih kola da rade savršeno pouzdano. Drugim rečima, dozvoljava se da frakcija of α_γ MAJ logičkih kola bude nepouzdana, dok ostatak od $(1-\alpha_\gamma)n$ MAJ logičkih kola radi pouzdano. Primetiti da ne uvodimo nikakave restrikcije vezane za nepouzdanost XOR kola. Kao i u prethodnom slučaju, dozvoljava se da sva otkazuju uvek kada njihov izlaz menja vrednost. Teorema data u nastavku opisuje performanse memorije pod navedenim uslovima.

Teorema 7.2. *Posmatrana memorijska arhitektura bazirana na* $(\gamma, \rho, \alpha, (7/8+\epsilon)\gamma)$ *ekspander kodu može da sačuva sve skladištene informacije proizvoljno dug vremenski period ako je ispunjeno*

$$\alpha_m + \alpha_\gamma < 3\epsilon(3+8\epsilon)\alpha/4. \tag{7.20}$$

Dokaz: U trenutku t=0 kodna reč ekspander koda upisana je u memoriju. Memorijske ćelije ažurirane su u trenucima $\ell\tau$, $\ell > 0$, pomoću jedne iteracije BF algoritma. Neka je sa $V(t)$ označen skup korumpiranih memorijskih ćelija u trenutku t. Termin korumpiran odnosi

se na ćelije koje u posmatranom trenutku skladište pogrešnu binarnu vrednost. Ovaj broj je u trenutku pre prvog ažuriranja $|V(\tau - \delta_0)|$ ograničen sa

$$|V(\tau - \delta_0)| \leq n\alpha_m. \tag{7.21}$$

Nakon ciklusa ažuriranja imamo

$$|V(\tau + \delta_0)| \leq \beta\alpha_m n + \alpha_\gamma n, \tag{7.22}$$

gde je prema jednačini (4.15) $\beta = (1 - 8\epsilon)/2$. U vremenskom intervalu $(\tau, 2\tau)$ moguće je da otkaže dodatno $\alpha_m n$ memorijskih ćelija, što u najgorem slučaju povećava broj korumpiranih ćelija za $\alpha_m n$. Tada dobijamo

$$|V(2\tau - \delta_0)| \leq \beta\alpha_m n + \alpha_m n + \alpha_\gamma n. \tag{7.23}$$

Na osnovu jednačine (4.15) i prethodne diskusije za svako $\ell > 1$ važi

$$\begin{aligned} |V((\ell+1)\tau - \delta_0)| \leq \beta\Big(|V(\ell\tau - \delta_0)| + |V((\ell-1)\tau - \delta_0)|\Big) \\ + \alpha_\gamma n + \alpha_m n. \end{aligned} \tag{7.24}$$

Na osnovu prethodne nejednakosti sleduje da se broj korumpiranih memorijskih ćelija može ograničiti sa gornje strane, što je predstavljeno lemom datom u nastavku.

Lema 7.4. *Broj korumpiranih memorijskih ćelija neposredno pre ℓ-tog ciklusa ažuriranja, $|V(\ell\tau - \delta_0)|$, za svako $\ell > 0$, zadovoljava*

$$|V(\ell\tau - \delta_0)| \leq (\alpha_m n + \alpha_\gamma n)/(8\epsilon). \tag{7.25}$$

Dokaz: Pogledati Dodatak 7.A ■

Kako prema formulaciji teoreme (7.20) sledi

$$(\alpha_m n + \alpha_\gamma n)/(8\epsilon) < (3 + 8\epsilon)\alpha n/4, \tag{7.26}$$

na osnovu analize date u [33], sledi da broj korumpiranih memorijskih ćelija u proizvoljno izabranom vremenskom trenutku nije veći od korektivne sposobnosti savršeno pouzdanog BF dekodera, pa je informacioni sadržaj sačuvan. Time je teorema dokazana. ■

Uslov izveden u prethodnoj teoremi ne govori ništa o prirodi otkaza u memorijskim ćelijama. Oni mogu biti nekorelisani ili korelisani, pri čemu je jedino važno da se njihov broj

između ciklusa ažuriranja ograniči. Otkazi mogu biti i permanentni s tom razlikom što tada α_m označava frakciju otkaza koji se dešavaju između zamena memorijskih ćelije koje su otkazale novim ćelijama. Ipak, u nastavku će biti smatrano da su otkazi memorijskih ćelija nekorelisani, što omogućava dokaz stabilnosti memorijske arhitekture.

Značajno je primetiti da dokaz pouzdanosti memorije u analizi prezentovanoj u [24] (Teorema 7.1) podrazumeva da se broj otkaza XOR kola ograničava. Ovde je pokazano da navedeni uslov nije potreban i u slučaju korelisanih otkaza logičkih kola veći broj nepouzdanih komponenti može biti upotrebljen u memorijskoj arhitekturi.

Sada će rezultat Teoreme 7.2 biti proširen na probabilistički model otkaza. Neka su $\Delta_m > 0$ i $\Delta_\gamma > 0$ takva da važi $p_m + \Delta_m = \alpha_m$ and $p_\gamma + \Delta_\gamma = \alpha_\gamma$. Tada, kada je uslov dat u jednačini (7.20) zadovoljen, može se formulisati sledeća lema.

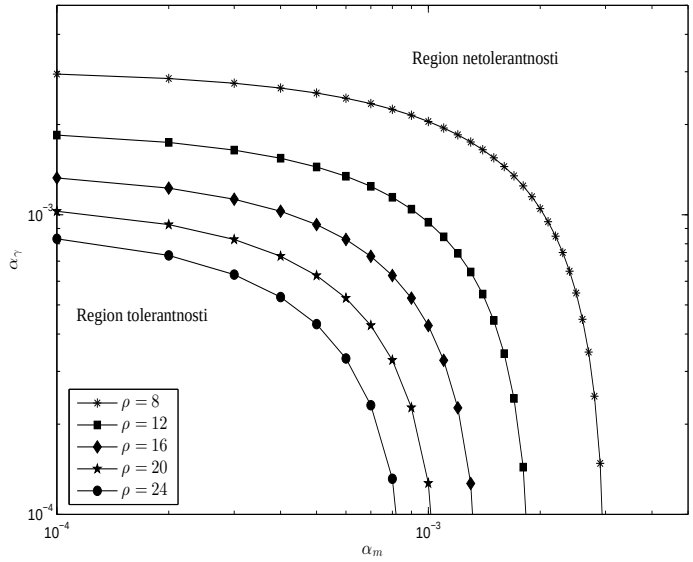
Lema 7.5. *Verovatnoća otkaza memorije nakon L ciklusa ažuriranja, $P_f(L)$, ograničena je sa*

$$P(L)_f \leq L(e^{-2\Delta_m^2 n} + e^{-2\Delta_\gamma^2 n}). \tag{7.27}$$

Dokaz: Dokaz sledi iz zapažanja da je prema *Chernoff*-ovoj granici verovatnoća da otkaže više od fiksne frakcije komponenata u vremenskom intervalu τ ograničena. ■

Prethodna lema opisuje veoma "labavu" granicu performansi i njen glavni cilj je da pokaže da verovatnoća otkaza memorije $P_f(L)$ opada eksponencijalno sa povećanjem dužine koda. Ona dokazuje postojanje memorije koja može da sačuva sve informacije u asimptotskom slučaju, kada se otkazi logičkih kola smatraju vremenski korelisanim.

Gornja granica desne strane nejednakosti (7.20), označena sa $3\epsilon(3 + 8\epsilon)\alpha/4$, numerički je određena i prikazana na slici 7.5. Pritom je za dobijnje gornje granice korišćena Lema 4.9, za fiksnu vrednost parametra ρ. Granične vrednosti raspodeljene su na α_m i α_γ, što je dovelo do formiranja regiona prikazanih na slici 7.5. Uočljivo je da se sa povećanjem ρ, podrazumevajući fiksno γ, broj suseda skupa od αn varijabilnih čvorova smanjuje. S druge strane, zahtevano je da svaki skup od αn varijabilnih čvorova ima ekspanziju od makar $7\gamma/8$, što se jedino može postići smanjujući vrednost α, što za posledicu ima da je vrednost gornje granice inverzno proporcijonalna sa ρ. Na primer, ako je $\rho = 8$ maksimalna frakcija komponenata koju je moguće tolerisati iznosi 0,003, dok za $\rho = 24$ dobijamo frakciju od 0,0009 otkaza komponenti.

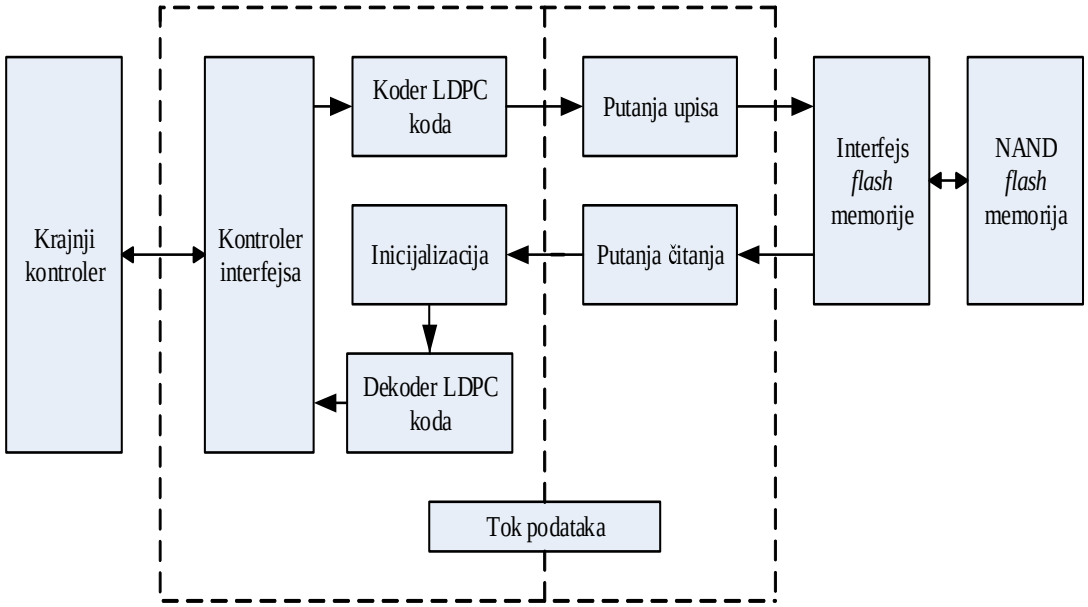


Slika 7.5: Regioni pouzdanosti memorijske arhitekture.

7.4 Potencijalna primena istaživanja na praktično značajne memorije

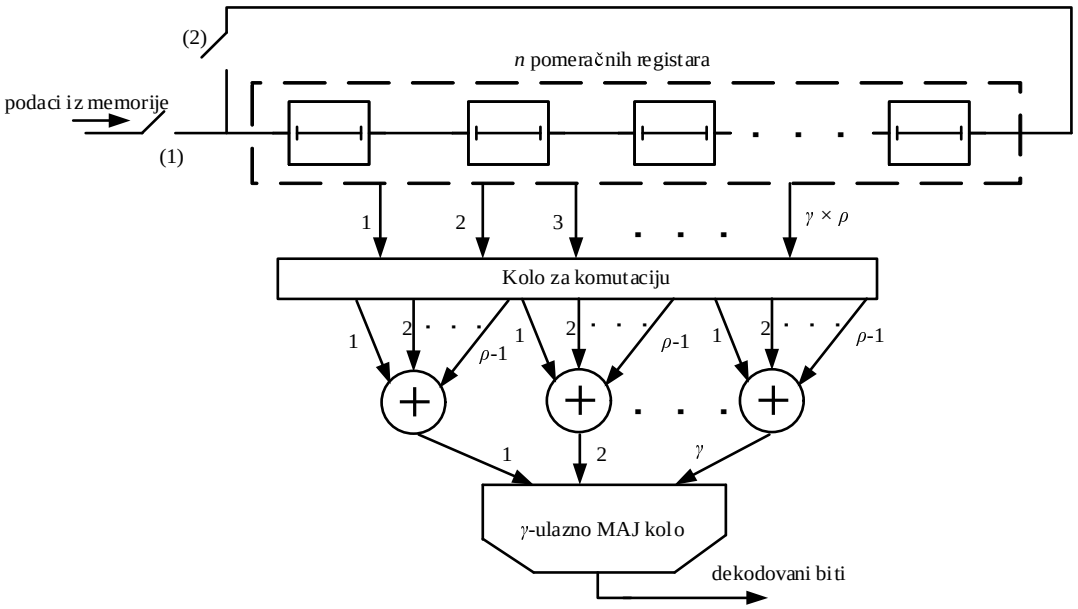
Iako je teorijski koncept kodovane memorije bazirane na LDPC kodovima u informacionom smislu veoma efikasn, njegova relativno velika kompleksnost onemogućava primenu u mnogim sistemima gde se zahveta kratko vreme pristupa memoriji, ili postoje restrikcije vezane za površinu dodeljenu redundantnim memorijskim ćelijama. Sistemi gde je moguće primeniti LDPC kodove vezani su pre svega za *flash* memorije, kao i različite trodimenzionalne memorijske arhitekture.

Problem pouzdanosti *flash* memorijskih uređaja postaje sve izraženiji sa većim faktorom integracije poluprovodničkih struktura. Uzroci nepouzdanosti *flash* memorija vezani su za efekte koji se javljaju prilikom čitanja iz memorije ili upisa u memoriju, kao i zbog istrošenosti memorijskih ćelija. Smatralo se da je verovatnoća da pročitana sekvenca iz NAND *flash* memorije, izgrađene u 90nm ili 65nm tehnologiji, bude neispravno dekodovana jednostavnim *Hamming*-ovim kodom u opsegu $10^{-16} - 10^{-13}$, što je omogućavalo njen uspešan rad [147]. Međutim, sa smanjenjem tehnologije na 45nm strukture (i manje), *Hamming*-ovi kodovi sa sposobnošću ispravljanja samo jedne greške na bloku od nekoliko stotina bita postaju neadekvatni. Slično, novije tehnolije izrade NOR *flash* memorije dovode do inherentne nepouzdanosti većeg nivoa nego što je to prihvatljivo. Nekoliko rešenja koja uključuju BCH ili *Reed–Solomon*-ove linearne



Slika 7.6: Blok dijagram kontrolera za zaštitu informacija *flash* memorija.

blok kodove, skorije predložena u [148–150] takođe ne pružaju odgovarajuću pouzdanost ovih memorija, zbog čega su najnovija istraživanja usmerena na upotrebu LDPC kodva [151–153]. Mane LDPC kodova, koje ih čine teško primenljivim u tradicionalnim RAM (eng. *Random Access Memory*) operativnim memorijama koje skladište pakete dužine 32 ili 64 bita, povezane sa velikom dužinom kodnih reči i komplikovanim iterativnim dekoderima, nisu prepreka njihove primene u NAND *flash* memorijama. Zahtevano vreme čitanja podataka iz memorije meri se desetinama mikrosekundi [148], što je dovljno za efikasnu implementaciju iterativnih dekodera koji razmenjuju meke informacije (kao što su *min-sum* ili različiti FAID dekoderi), čak i kada su kodne reči duge i po više hiljada bita. S druge strane, korektivne sposobnosti FAID dekodera nisu do kraja poznate, niti su poznati efekti koji se javljaju prilikom implementacije dekodera na nepouzdanom hardveru. Upotreba jednostavnijih dekodera kao što su OS-MAJ ili *bit-flipping* dekoderi, analizirani u Poglavlju 4, daje dobar odnos kompeksnosti i pouzdanosti, posebno kada se posmatraju samo memorijske ćelije sa dva nivoa naboja (eng. *single level cell*). Predložena rešenja *flash* memorija bazirana na LDPC kodovima konstruisanim pomoću Euklidske geometrije [154, 155] deluju podsticajno za istraživanje ekspander kodova veće dužine i komplesnosti, čije performanse će biti procenjene metodama predloženim u Poglavlju 4. Treba naglasiti da *Taylor*-ov sistem periodičnog ažuriranja nije primenljiv na *flash* memorije, zbog njihove osetvljivosti na višestruke pristupe, već se informacija dekoduje samo onda kada se čita iz memorije. Umesto toga, kako je to prikazano na slici 7.6, proces kon-



Slika 7.7: Arhitektura kola za korekciju grešaka u *flash* memoriji.

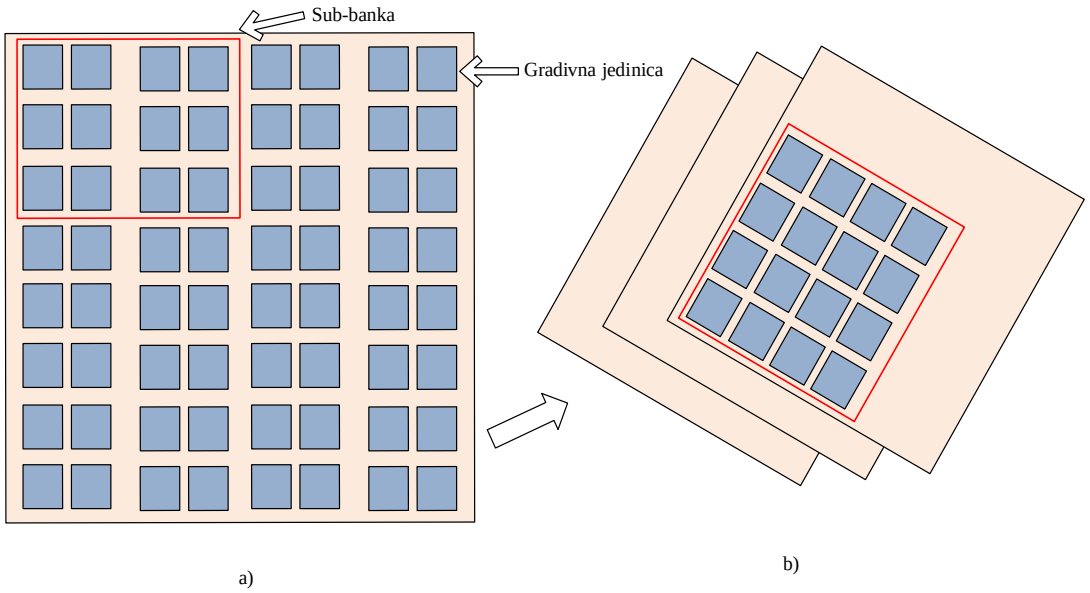
trolne grešaka izmešta se van memorijskog uređaja i pristupa mu se samo u okviru operacionih ciklusa memorije. Analiza prezentovana u Poglavlju 4, pokazala je viskok stepen sigurnosti koju pružaju OS-MAJ dekoderi konstruisani delimično od nepouzdanih komponenti, kada zaštitu obavljaju LDPC kodovi konstruisani na osnovu konačnih geometrija. Tako je pokazano da je verovatnoću greške nakon čitanja podatake iz memorije $< 10^{-12}$ moguće dobiti ako bi se iskoristio $PG(2, 2^4)$ kod, čak i za ekstremno visoku nepouzdanost memorijskih ćelija od 10^{-3}. S druge strane, nije teško pokazati da bi u istom slučaju *Hamming*-ovi kodovi mogli da ponude zaštitu ne manju od 10^{-5}. Predloženi kod je relativno kratak (dužina koda je 273 bita), visokog kodnog količnika koji iznosi približno 0,7, dok se dekodovanje obavlja u jednom koraku i ne stvara nepotrebno kašnjenje. Zahtev za većom pouzdanošću moguće je ispuniti upotrebom nekog dužeg PG, AG ili EG koda.

Operacije dekodovanja mogu se izvršiti u paraleli, ili, ako je potrebno smanjiti kompleksnost dekodera, dekodovanje je moguće obaviti i serijski, kako je to prikazano na slici 7.7. U n pomeračkih registara prvo se upisuju kodni biti, što se postiže zatvaranjem prekidača (1) i otvaranjem prekidača (2). Nakon toga se pristupa dekodovanju (zatvara se prekidač (2), a otvara prekidač (1)), pri čemu se u svakom intervalu takta dekoduje jedan kodni bit. Kolo za komutaciju povezuje izlaze iz registara sa ulazima XOR kola, pri čemu su ove veze fiksne za zadati kod i ne zavise od bita koji se trenutno dekoduje. Osobina cikličnosti koju imaju kodovi konstruisani na osnovu konačnih geometrija omogućava da se izvrši ciklični pomeraj kodnih

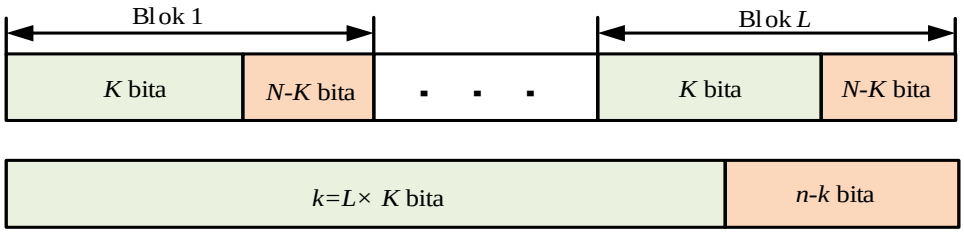
bita (povratna sprega), a da se veze u komutatoru ne menjaju. Tako je potrebno implementirati samo jedno γ-ulazno MAJ kolo i γ XOR kola sa $\rho - 1$ ulaza, pri čemu je XOR kola moguće napajati i nižim naponom od nominalnog, kako je to pokazala analiza prezentovana u Poglavlju 4.

Procesorske memorije izgrađene od trodimenzionalnih (3D) poluprovodničkih struktura predstavljaju potencijalno interesantno rešenje za *Taylor*-ov memorijski koncept. Preciznije reč je polihedralnoj memorijskoj arhitekturi koju je predložio *Lefter* [156]. Memorije većeg kapaciteta obično su hijararhijski organizovane, tako da se adresiranje i rutiranje u memoriji obavlja na nivou *memorijskih banaka*. Jedna memorijska banka, predstavljena na slici 7.8.a, dalje je podeljena na *sub-banke*, pri čemu se memorijski pristup (čitanje ili upisivanje) obavlja na jednoj sub-banci. Sub-banke se dalje dele na manje gradivne jedinice, koje mogu biti kodne reči nekog linearnog blok koda. Polihedralne 3D memorije imaju i vertikalnu komponentu, a konstruišu se tako da se horizontalne memorijske banke slažu jedna na drugu (slika 7.8.b). Ono što predstavlja osnovnu prednost 3D arhitektura u odnosu na tradicionalne 2D memorije ogleda se u višestrukom pristupu ovim memorijama. Dok se upisivanje/čitanje 2D memorija može obavljati samo u okviru jedne sub-banke u jednom operacionom ciklusu, 3D memorije omogućavaju i vertikalni pristup memoriji, tj. moguće je vršiti upis ili čitanje iz memorije kroz više sub-banaka koje pripadaju različitim bankama (ravnima u memoriji). Upravo ovakva arhitektura omogućava da se više osnovnih gradivnih jedinica kombinuju u jedinstvene kodne reči, koje će se zatim periodično čitati iz memorije, propustiti kroz kolo za korekciju grešaka, a zatim opet upisati na iste memorijske lokacije. Mogućnost višestrukog memorijskog pristupa obezbeđuje da se proces ažuriranja obavalja nezavisno od procesa čitanja/upisivanja u memoriju. Kolo za ažuriranje memorijskih lokacija, kao i LDPC koder i dekoder koji su potrebni pri upisu, odnoso čitanju podataka mogu se smestiti na jednom od horizontalnih silikonskih slojeva. Pri tome je brzina operacija u memoriji dovoljna da se obave procesi kodovanja i dekodovanja LDPC kodova dužina nekoliko stotina bita. Treba naglasiti da je proces ažuriranja funkcionalno sličan osvežavanju naponskih nivoa DRAM (eng. *Dynamic RAM*) memorijske arhitekture, s tom razlikom što se u DRAM arhitekturi obavlja samo čitanje i upisivanje, bez korekcije grešaka.

Da bi LDPC kodovi mogli zameniti *Hammnig*-ove kodove u operacionim 3D memorijama, potrebno je preskalirati vrednosti blokova koji se koduju. Kako su kodne reči LDPC kodova tipično duge nekoliko stotina (pa i hiljada) bita, to je blokove od 16 ili 32 bita (koliko iznose



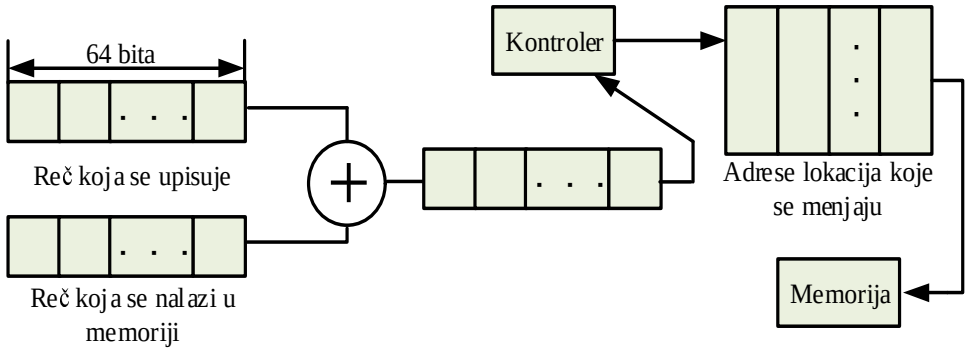
Slika 7.8: Hijararhijska memorijska arhitektura: a) 2D memorija; b) 3D memorija.



Slika 7.9: Formiranje okvira za primenu LDPC koda.

dužine kodnih reči tradicionalnih *Hamming*-ovoh kodova) potrebno grupisati uz zadržavanje istog (ili sličnog) kodnog količnika, kako je to ilustrovano na slici 7.9. Potencijalno interesantno rešenje opet predstavljaju kodovi bazirani na konačnim geometrijama, koji ostvaruju visoke kodne količnike i izuzetne korektivne sposobnosti. Prelazak na LDPC kodove i *Taylor*-ovu arhitekturu otvara niz problema koje treba rešiti. Na ovom mestu će biti skrenuta pažnja na nekoliko najznačajnijih problema, čija moguća rešenja će takođe biti predložena.

- *Obrađivanje dugih kodnih reči zahteva kompikovan mehanizam rutiranja poruka i redundansu u vidu kodera i dekodera LDPC kodova koju je potrebno implementirati.* Povećana redundansa usled obrađivanja dugih kodnih reči može biti prepreka njihovoj komercijalizaciji, ali konstantno smanjenje tehnologija omogućava implementaciju kompleksnije logike na fiksnoj silikonskoj površini. Specifičnost 3D memorija posebno pogoduje upotrebi LDPC kodova, jer je moguće čitav jedan horizontalni sloj dodeliti koderu i dekoderu. Dodatno, ako se izabere arhitektura koja koristi OS-MAJ dekoder kao kolo za

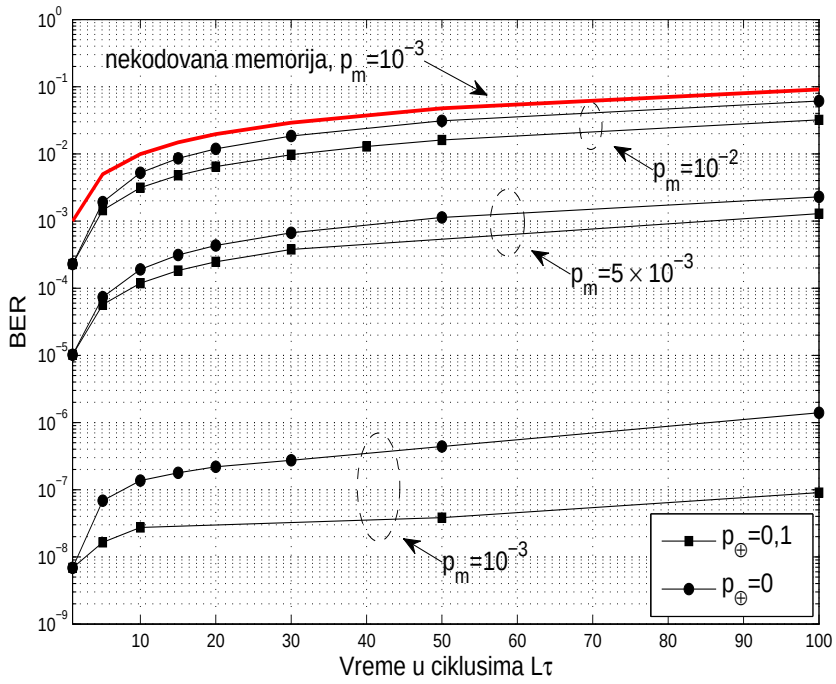


Slika 7.10: Ilustracija upisa podataka u memorije bazirane na LDPC kodovima.

ažuriranje (slika 7.7), tada je potrebno implementirati svega nekoliko XOR kola i jedno MAJ kolo. Višestruki pristupi memorijskim lokacijama olakšavaju proces rutiranja, jer je moguće grupisati podatke iz različitih ravni memorije u jedinstvene kodne reči.

- *Uvođenje principa ažuriranja dovodi do mogućne kolizije između podataka koji se ažuriraju i podataka koji se čitaju ili upisuju u memoriju.* Ažuriranje memorijskih lokacija zahteva dizajniranje kontrolera koji bi rukovodio rasporedom lokacija koje se ažuriraju. Vreme za koje se potrebno ažurirati sve memorijske lokacije zavisi direktno od nepouzdanosti memorije, kao i od njene veličine. Više pouzdane memorije omogućavaju ređe ažuriranje, pa samim tim smanjuju verovatnoću kolizije. Kako se smatra da su podaci nakon ciklusa ažuriranja pouzdaniji nego pre ciklusa, to bi bilo optimalno dati prednost ažuriranju nego čitanju podataka iz memorije. Dodatno, ažuriranje je potrebno obaviti samo onda kada ima grešaka, pa se može konstruisati arhitektura koja će periodično proveravati da li su sve provere parnosti zadovoljene, a onda samo u slučaju da nisu ažurirati delove memorije za koje se zna da sadrže greške.

- *Komplikovana obrada povećava vreme pristupa memoriji.* Koder i dekoder LDPC kodova su znatno složeniji od komponenata *Hamming*-ovog koda, ali se efikasnim pristupom kompleksnost može redukovati. Ako se implementira kod u sistematskom obliku, prilikom čitanja podatka iz memorije nije potrebno vršiti dodatno dekodovanje, već samo pročitati vrednosti lokacija koje sadrže informacione bite. Tada vreme čitanja nije degradirano. S druge strane, upisivanje u memoriju zahteva dodatne operacije, koje obuhvataju računanje kontrolnih bita prema algoritmu prezentovanom u Odeljku 3.2.4. Međutim, kako se u jednom trenutku upisuje samo mali deo informacionih bita nema potrebe za računanjem svih provera parnosti, već samo onih u kojima biti koji se menjaju učestvuju.



Slika 7.11: Performanse memorije bazirane na $PG(2, 2^3)$ kodu.

Moguće je dizajnirati protokol koji će optimizovati broj pristupa memoriji i koji uključuje skladištenje informacija o proverama parnosti za svaki informacioni bit. Tako se u posebnim registrima čuvaju adrese kontrolnih bita koje treba proveriti. Moguće je prvo pročitati niz informacionih bita iz lokacije na koju treba upisati podatke, pronaći sve binarne vrednosti koje se razlikuju i samo promeniti kontrolne bite koji će se razlikovati nakon upisa informacija. Ovo je ilustrovano na slici 7.10. Treba istaći da je pronalaženje efiksne organizacije kodovane memorije problem za sebe. U ovom radu akcenat je pre svega na pouzdanosti memorija, pa su problemi memorijske organizacije ostavljeni za buduća istraživanja.

Na slici 7.11 ilustrovane su performanse memorije bazirane na BF dekoderu i $PG(2, 2^3)$ kodu, izražene preko zavisnosti verovatnoće greške po bitu (BER) od broja ciklusa ažuriranja (L), za različite vrednosti verovatnoće otkaza memorijskih ćelija i XOR logičkih kola, p_m i $p_\oplus$, respektivno. Prednosti upotrebe kodovane memorije primetne su čak i ako su ekstremno nepouzdana XOR kola ($p_{XOR} = 0, 1$) korišćena pri realizaciji. Tako na primer, ako je $p_m = 10^{-3}$, verovatnoću greške manju od 10^{-6} moguće je održavati do 100 ciklusa ažuriranja. S druge strane, ako se ne bi koristio PG kod, svaki deseti informacioni bit bi bio pogrešno pročitan iz memorije. Takođe, uočava se da se pouzdanost kodovane memorije naglo (nelin-

earno) povećava sa smanjenjem nepouzdanosti memorijskih ćelija. Treba napomenuti da se verovatnoća greške može dodatno smanjiti ako se, pre korišćenja, sadržaj memorije dodatno dekodeuje nekim pouzdanim iterativnim dekoderom, kako je to originalno predložio *Taylor*.

7.5 Zaključak

U ovom poglavlju izložene su osnove kodovanih memorija baziranih na LDPC kodovima. Određene su redundanse memorija koje za korekciju grešaka koriste *hard-decision* princip dekodovanja. Niska kompleksnost, visoki kodni količnici i garantnovano ispravljanje grešaka glavni su razlozi upotrebe *Hamming*-ovih kodova u praktičnim realizacijama memorijskih uređaja. Međutim, iako nisu pogodni za primene koje zahtevaju kratke dužine kodnih reči, LDPC kodovi su upotrebljivi ako primena zahteva duže kodove, jer kompleksnost memorije bazirane na LDPC kodovima raste linearno sa povećanjem kodne dužine, što nije slučaj ako se primene drugi linearni blok kodovi. Dodatno, veliki izbor kodnih količnika, kao i parametara koda povećava njihov značaj u memorijskim uređajima. Iako je za male vrednosti parametara γ i g broj otkaza koji garantovano neće ugroziti rad memorije relativno mali, performanse koje postižu ovakve memorije prevazilaze uređaje u kojima su implementirani *Hamming*-ovi kodovi. Pri tome treba naglasiti da *Hamming*-ovi kodovi nisu otporni na otkaze u logičkim kolima, koji mogu značajno da degradiraju performanse memorije. Upotreba LDPC kodova sa većim vrednostima g ili γ moguća je u primenama gde kompleksnost nije previše kritična. Jedna od mogućih praktičnih primena svakako obuhvata *flash* memorije novije generacije.

I pored skorijeg pokušaja da se vrednost memorijskog kapaciteta ograniči sa gornje strane [157], najznačajniji teorijski problem vezan za pouzdanost kodovanih memorija ostaje preciznija karakterizacija memorijskog kapaciteta. S druge strane, detaljniji uvid u činjenicu da otkazi logičkih kola mogu poboljšati performanse dekodovanja može dovesti do memorijskih arhitektura koje su pouzdanije i otpornije na potencijalne otkaze memorijskih ćelija.

Dodatak 7.A (dokaz Leme 7.4)

Broj korumpiranih ćelija pre ℓ-tog ciklusa ažuriranja zadovoljava

$$|V(\ell\tau - \delta_0)| \leq A_1\lambda_1^{-\ell} - A_2\lambda_2^{-\ell} + K, \tag{7.28}$$

gde je $\lambda_1 = -0.5(1 + \sqrt{1 + 4/\beta})$, $\lambda_2 = 0.5(\sqrt{1 + 4/\beta} - 1)$, konstante A_1 i A_2 predstavljaju univerzalno rešenje diferencne jednačine $x_\ell - \beta x_{\ell-1} - \beta x_{\ell-2} = \alpha_m n + \alpha_\gamma n$, čije partikularno rešenje K zadovoljava inicijalne uslove $x_1 = \alpha_m$ i $x_2 = (\beta+1)\alpha_m n + \alpha_\gamma n$. Nije teško pokazati da važi

$$A_j = \frac{\beta(2 + \lambda_j)\alpha_m n + (1 + \beta\lambda_j)\alpha_\gamma n}{\beta(\lambda_2 - \lambda_1)(1 - 2\beta)}, \quad j = 1, 2, \tag{7.29}$$

i

$$K = (\alpha_m + \alpha_\gamma)n/(1 - 2\beta) = (\alpha_m + \alpha_\gamma)n/(8\epsilon). \tag{7.30}$$

Kako je $|A_1/A_2| < 1$ i $2 > |\lambda_1/\lambda_2| > 1$, desna strana izraza (7.28) monotono raste sa ℓ, pa se konačno dobija

$$\lim_{\ell\to\infty}\left[A_1\lambda_1^{-\ell} - A_2\lambda_2^{-\ell} + K\right] = K. \tag{7.31}$$

Poglavlje 8

Generalni zaključak i predlog budućih istraživanja

Ovaj rad obrađuje temu pouzdanog prenosa i čuvanja informacija pomoću logičkih kola napravljenih od nepouzdanih hardverskih komponenti. Motivisan je sve izraženijom nepouzdanošću elektronskih uređaja, konstruisanim u novim energetski-efikasnim CMOS tehnologijama, i potrebom da se nepouzdanost kontroliše upotrebom zaštitnih kodova. Intezivan razvoj LDPC kodova u protekle dve decenije učinio ih je najrelevantnijom klasom linearnih blok kodova, koja dostiže *Shanonn*-ov kapacitet na binarnim kanalima. Potencijalna primena LDPC kodva za povećanje pouzdanosti sistema napravljenih od nepouzdanih komponenti u literaturi već je ustanovljena, kroz seriju relevantnih članaka koja razmatra asimptotsko ponašanje iterativnih dekodera LDPC kodova. Međutim, iako je primećena robusnost ovog tipa dekodovanja na otkaze komponentnih logičkih kola, prethodni rad nije pružio odgovore na fundamentalna pitanja: *koliko je izražena degradacija performansi dekodera usled nepouzdanosti pojedinih komponenti* i *da li je moguće garantovati ispravljanje određenog broja greška.* Ovaj rad pruža inovativne odgovore na ta pitanja uz uvođenje realističnijeg modela otkaza logičkih kola od onog koji se dominantno koristi u literaturi.

Tako je pokazano da se koncept garantovanog ispravljanja grešaka može proširiti i na dekodere sastavljene od nepouzdanih komponenata. Na primeru jednostavnog *bit-flipping* dekodera ilustrovana je osobina nepouzdanih dekodera da ispravljaju fiksnu frakciju kanalnih grešaka. Ova osobina je važna jer omogućava dizajniranje LDPC koda koji ima sposobnost da ispravi zadati broj grešaka uz zadržavanje konstantne kompleksnosti i kodnog količnika (uz uslov povećanja dužine koda), što je osobina koju ima samo mali broj dekodera linearnih

blok kodova, realizovanih od pouzdanih komponenti. Imajući u vidu sličnost u metodologiji dokazivanja, može se intuitivno pretpostaviti da neki *message-passing* dekoderi (kao na primer *Gallager B* ili *min-sum* dekoder) imaju iste korektivne osobine, ali su formalni dokazi ostavljeni za budući rad.

Gallager B dekoder je u ovom radu posmatran iz ugla koji pruža drugačiji uvid u problem nepouzdanosti – testirane su performanse praktično značajnih kodova i predložena modifikacija koja garantuje visok stepen otpornosti na otkaze logičkih kola. Posebno su analizirai uslovi koji dovode do poboljšanja performansi dekodera kao posledice unošenja grešaka u dekoder. Otkriće da korelisani otkazi logičkih kola, ne samo da mogu smanjiti nivo zaostale greške, već mogu i ubrzati proces dekodovanja iskorišćena je za konstrukciju hibridnog dekodera čije performanse prevazilaze performanse većine znatno složenijih rešenja. Dodatno, predloženi koncept razmene poruka (MAE dekoder) u okviru grafa otvara novi pravac istraživanja, u kome se poruke više ne računaju identično u svakoj iteraciji, već se dozvoljava promena lokalnog pravila odlučivanja u zavisnosti od iteracije dekodovanja. MAE dekoder je prvi korak ka stvaranju "kognitivnog dekodera" koji bi učio na iskustvu i prilagođavao lokalne odluke prema konkretnoj situaciji.

Iako je poznata činjenica da dekoderi LDPC kodova predstavljaju teorijski optimalno rešenje za čuvanje informacija u nepouzdanim memorijskim ćelijama, u praksi su pre svega zbog izražene jednostavnosti prednost dobijali *Hamming*-ovi, BCH ili *Reed-Solomon*-ovi kodovi. Međutim, trend povećanja nepouzdanosti kako memorijskih tako i kombinacionih logičkih elemenata vodi ka potpunoj neupotrebljivosti dosadašnjih rešenja. Kako je to pokazala studija prezentovana u ovom radu memorijske arhitekture bazirane na *bit-flipping* dekoderu imaju sposobnost tolerisanja fiksne frakcije komponenata, čak i kada su otkazi vremenski korelisani. Sličan rezultat u literaturi nije poznat ni za jednu memoriju kodovanu drugim zaštitinim kodom. Dizajniranje brzih trodimenzionalnih memorija sa mogućnošću višestrukog pristupa, otvara novo polje praktične primene teorijskih koncepata, poput memorijske organizacije razmatrane u ovom radu.

Bibliografija

[1] G. Varatkar, S. Narayanan, N. Shanbhag, and D. Jones, "Stochastic networked computation," *IEEE Trans. VLSI Syst.*, vol. 18, no. 10, pp. 1421–1432, Oct. 2010.

[2] Y. Ye, S. Gummalla, C. Wang, C. Chakrabarti, and Y. Cao, "Random variability modeling and its impact on scaled CMOS circuit," *ACM J. of Comput. Electron.*, vol. 9, no. 3, pp. 108–113, 2010.

[3] J. Chen, C. Spagnol, S. Grandhi, E. Popovici, S. Cotofana, and A. Amaricai, "Linear compositional delay model for the timing analysis of sub-powered combinational circuits," in *Proc. of IEEE Comp. Soc. Annual Symp. on VLSI*, July 2014.

[4] M. Tanner, D. Sridhara, A. Sridharan, T. Fuja, and D. Costello, "LDPC block and convolutional codes based on circulant matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, p. 2966–2984, Dec. 2004.

[5] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, p. 599–618, Feb. 2001.

[6] http://www.itrs2.net/.

[7] http://www.nist.gov/pml/semiconductor/cmos/cmos.cfm.

[8] S. Ghosh and K. Roy, "Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1718–1751, Oct. 2010.

[9] C. Shannon, "Mathematical theory of communication," *Bell System Tech Journal*, vol. 27, pp. 379–423, Sep. 1948.

[10] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding," in *Proc. Int. Communications Cconf. (ICC'93)*, Geneva, Switzerland, May 1993, p. 1064–1070.

[11] D. MacKay, "Good error-correctingcodes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 2, no. 45, pp. 399–341, Mar. 1999.

[12] B. Vasic and S. K. Chilappagari, "An information theoretical framework for analysis and design of nanoscale fault-tolerant memories based on low-density parity-check codes," *IEEE Transactions on Circuits and Systems I, Regular Papers*, vol. 54, no. 11, pp. 2438–2446, Nov. 2007.

[13] J. Von Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," in *Automata Studies*, C.E. Shannon and J. McCarty, eds., Princeton Univ. Press, July 1956, pp. 43–98.

[14] H. Jie and P. Jonker, "A system architecture solution for unreliable nanoelectronic devices," *IEEE Transactions on Nanotechnology*, vol. 1, no. 4, pp. 201–208, Dec. 2002.

[15] S. Roy and V. Beiu, "Majority multiplexing-economical redundant fault-tolerant designs for nanoarchitectures," *IEEE Transactions on Nanotechnology*, vol. 4, no. 4, pp. 441–451, July 2005.

[16] H. Goronkin and Y. Yang, "High-performance emerging solid-state memory technologies," *MRS Bulletin, special issue on High-Performance Emerging Solid-State Memory Technologies*, vol. 29, no. 11, pp. 805–813, Nov. 2004.

[17] M. Taylor, "Reliable computation in computing systems designed from unreliable components," *Bell System Technical Journal*, vol. 47, pp. 2339–2366, 1968.

[18] P. Elias, "Computation in the presence of noise," *IBM Journal of Research and Development*, vol. 2, no. 4, pp. 346–353, Oct. 1958.

[19] R. Dobrushin and S. Ortyukov, "Upper bound on the redundancy of self-correcting arrangements of unreliable functional elements," *Problemy Peredachi Informatsii*, vol. 13, no. 3, pp. 82–89, 1958.

[20] S. Winograd and J. D. Cowan, *Reliable Computation in the Presence of Noise*. Cambridge, MA, USA: MIT Press, 1963.

[21] N. Pippenger, "Developments in the synthesis of reliable organisms from unreliable gates," in *Proceedings of Symposia in Pure Mathematics*, 1990.

[22] M. Taylor, "Reliable information storage in memories designed from unreliable components," *Bell System Technical Journal*, vol. 47, pp. 2299–2337, 1968.

[23] A. Kuznetsov, "Information storage in a memory assembled from unreliable components," *Problems of Information Transmission*, vol. 9, pp. 254–264, 1973.

[24] S. Chilappagari and B. Vasic, "Fault tolerant memories based on expander graphs," in *Proceedings of IEEE Information Theory Workshop*, Tahoe City, CA, USA, 2-7 Sep. 2007, pp. 126–131.

[25] L. Varshney, "Performance of LDPC codes under faulty iterative decoding," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4427–4444, July 2011.

[26] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA, USA: MIT Press, 1963.

[27] F. Kschischang, B. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[28] J. Pearl, *Probablisitic Reasoning in Intelligent Systems*. San Francisco, CA: 2nd ed. Kaufmann, 1988.

[29] S. Lauritzen, B. Frey, and H. Loeliger, "Local computations with probabilities on graphical structures and their application to expert systems," *J. Roy. Statist. Soc., ser. B*, vol. 50, no. 2, p. 157–224, 1988.

[30] N. Wiberg, *Codes and decoding on general graphs*. Sweeden: Ph.D. dissertation, Dept. Elec. Eng, U. LinkSping, 1996.

[31] G. Forney, "On iterative decoding and the two-way algorithm," in *Proc. Int. Symp. Turbo Codes and Related Topics*, Brest, France, Sep. 1997.

[32] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, p. 619–637, Feb. 2001.

[33] M. Sipser and D. Spielman, "Expander codes," *IEEE Transactions on Information Theory*, vol. 42, no. 6, p. 1710–1722, Nov. 1996.

[34] A. Barg and G. Zemor, "Error exponents of expander codes," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, p. 1725–1729, June 2002.

[35] G. Zemor, "On expander codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 835–837, Feb. 2001.

[36] D. Burshtein and G. Miller, "Expander graph arguments for messagepassing algorithms," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 782–790, Feb. 2001.

[37] C. H. Huang and L. Dolecek, "Analysis of finite alphabet iterative decoders under processing errors," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Candada, May 2013, pp. 5085–5089.

[38] S. M. S. Tabatabaei Yazdi, H. Cho, and L. Dolecek, "Gallager B decoder on noisy hardware," *IEEE Transactions on Communications*, vol. 61, no. 5, pp. 1660–1673, May 2013.

[39] C. Kameni Ngassa, V. Savin, and D. Declercq, "Min-Sum-based decoders running on noisy hardware," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '13)*, Atlanta, USA, Dec. 2013, pp. 1–5.

[40] E. Dupraz, D. Declercq, B. Vasic, and V. Savin, "Finite alphabet iterative decoders robust to faulty hardware: Analysis and selection," in *8th International Symposioum on Turbo Codes and Iterative Information Processing (ISTC)*, Bremen, Germany, Aug. 2014, pp. 1–10.

[41] A. Balatsoukas-Stimming and A. Burg, "Density evolution for min-sum decoding of LDPC codes under unreliable message storage," *IEEE Communications Letters*, vol. 18, no. 5, pp. 849–852, May 2014.

[42] S. M. S. Tabatabaei Yazdi, C. H. Huang, and L. Dolecek, "Optimal design of a Gallager B noisy decoder for irregular LDPC codes," *IEEE Communications Letters*, vol. 16, no. 12, pp. 2052–2055, Dec. 2012.

[43] B. Vasic, D. Nguyen, and S. Chilappagari, *Chapter 6 - Failures and Error Floors of Iterative Decoders*. Oxford: Academic Press, 2014.

[44] H. Ando, R. Kan, Y. Tosaka, K. Takahisa, and K. Hatanaka, "Validation of hardware error recovery mechanisms for the signal probability SARC64 V microprocessor," in *Proc. Intl. Conf. Dependable Syst. and Networks*, 2008, pp. 62–69.

[45] S. Nassif, K. Bernstein, D. Frank, A. Gattiker, W. Haensch, B. Ji, E. Nowak, D. Pearson, and N. Rohrer, "High performance CMOS variability in the 65nm regime and beyond," in *Proc. Intl. Electron Devices Meeting*, 2007, pp. 569–571.

[46] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *IEEE Transactions on Computer-aided Design of Integreted Circuits and Systems*, vol. 24, no. 12, pp. 1859–1880, Dec. 2005.

[47] A. Bhavnagarwala, X. Tang, and J. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability," *IEEE J. Solid-State Circuits*, vol. 36, no. 4, p. 658–665, Apr. 2001.

[48] R. Micheloni, A. Marelli, and R. Ravasio, *Error Correction Codes for Non-Volatile Memories*. Springer Science+Business Media B.V., 2008.

[49] R. Rithe, J. Gu, A. Wang, S. Datla, G. Gammie, D. Buss, and A. Chandrakasan1, "Non-linear operating point statistical analysis for local variations in logic timing at low voltage," in *Proc. Europe Conference Design, Automation, Test and Exhibition (DATE)*, Dresden, Germany, Mar. 2010, pp. 965–968.

[50] A. Djahromi, A. Eltawil, F. Kurdahi, and R. Kanj, "Cross layer error exploitation for aggressive voltage scaling," in *Proc. Int. Symp. Quality of Electronic Design (ISQED 2007)*, San Jose, CA, USA, Mar. 2007, pp. 192–197.

[51] S. Zaynoun, M. S. Khairy, A. M. Eltawil, F. J. Kurdahi, and A. Khajeh, "Fast error aware model for arithmetic and logic circuits," in *Proceedings of 30th IEEE International Conference on Computer Design (ICCD)*, Montreal, QC, Sept.–Oct. 2012, pp. 322–328.

[52] R. Marculescu, D. Marculescu, and M. Pedram, "Probabilistic modeling of dependencies during switching activity analysis," *IEEE Trans. Computer-aided design of integreted circuits and systems*, vol. 17, no. 2, p. 73–83, Feb. 1998.

[53] A. Pirbadian1, M. Khairy, A. Eltawil, and F. Kurdahi, "State dependent statistical timing model for voltage scaled circuits," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, Melbourne, Australia, June 2014, pp. 1432–1435.

[54] A. Amaricai, S. Nimara, O. Boncalo, J. Chen, and E. Popovici, "Probabilistic gate level fault modeling for near and sub-threshold CMOS circuits," in *Proc. 17th Euromicro Conf. on Digital Syst. Design (DSD)*, Verona, Avg. 2014, pp. 473–479.

[55] N. Miskov-Zivanov and D. Marculescu, "A systematic approach to modeling and analysis of transient faults in logic circuits," in *Proc. Int. Symp. Quality of Electronic Design (ISQED 2009)*, San Jose, CA, USA, Mar. 2009, pp. 408–413.

[56] N. George, C. Elks, J. B., and J. Lach, "Transient fault models and avf estimation revisited," in *Proc. IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, Chicago, IL, USA, June-July 2010, pp. 477–486.

[57] S. Krishnaswamy, G. Viamontes, I. Markov, and J. Hayes, "Probabilistic transfer matrices in symbolic reliability analysis of logic circuits," *ACM Trans. Design Autom. Electron. Syst.*, vol. 13, no. 1, p. 8.1–8.35, Jan. 2008.

[58] K. Parker and E. McCluskey, "Probabilistic treatment of general combinational networks," *IEEE Trans. Comput.*, vol. C-24, no. 6, pp. 668–670, June 1975.

[59] S. Ercolani, M. Favalli, M. Damiani, P. Olivio, and B. Rico, "Estimate of signal probability in combinatorial logic networks," in *Proc. of European Test Conference*, Paris, France, Apr. 1989, pp. 132–138.

[60] J. Savir, G. Ditlow, and P. Bardell, "Random pattern testability," in *Proc. of IEEE Symposium on Fault Tolerant Computing*, Milan, Italy, June 1983, pp. 80–89.

[61] S. Brkić, P. Ivaniš, G. Đorđević, and B. Vasić, "Symbolic analysis of faulty logic circuits under correlated data-dependent gate failures," *Telfor Journal*, vol. 6, no. 1, pp. 2–6, 2014.

[62] S. Brkic, , P. Ivanis, and D. G., "Taylor-Kuznetsov fault-tolerant memories: a survey and results under correlated gate failures," in *Proc. IEEE TELSIKS 2013*, Nis, Serbia, Oct. 2013, pp. 455–462.

[63] A. Abdollahi, "Probabilistic decision diagrams for exact probabilistic analysis," in *Proc. Intl. Conf. Comput.-Aided Design*, San Jose, USA, Nov. 2007, pp. 266–272.

[64] G. Asadi and M. Tahoori, "An analytical approach for soft error rate estimation in digital circuits," in *Proc. Intl. Symp. Circuits and Syst.*, May 2005, pp. 2991–2994.

[65] C. Yu and J. Hayes, "Trigonometric method to handle realistic error probabilities in logic circuits," in *Proc. Design, Automation and Test in Europe*, Grenoble, France, Mar. 2011, pp. 1–6.

[66] S. Luckenbill, J. Lee, Y. Hu, R. Majumdar, and L. He, "Ralf: Reliability analysis for logic faults – an exact algorithm and its applications," in *Proc. Design, Automation and Test in Europe*, Dresden, Germany, Mar. 2010, pp. 783–788.

[67] C.-C. Yu, *Probabilistic Analysis for Modeling and Simulating Digital Circuits*. Doktrorska disertacija, 2012, [online] https://deepblue.lib.umich.edu/handle/2027.42/93816.

[68] T. Rejimon and T. Bhanja, "Scalable probabilistic computing models using bayesian networks," in *Proc. Midwest Symp. Circuits and Syst*, Covington KY, Aug. 2005, pp. 712–715.

[69] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. 27, no. 5, pp. 533–547, May 1981.

[70] T. Nozaki, "Parallel encoding algorithm for LDPC codes based on block-diagonalization," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2015, p. 1911–1915.

[71] B. A. and G. Zemor, "Codes on hypergraphs," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Toronto, Canada, July 2008, pp. 156–160.

[72] Y. Bilu and S. Hoory, "On codes from hypergraphs," *European Journal of Combinatorics*, vol. 25, p. 339–354, 2004.

[73] C. Kelley and D. Sridhara, "Eigenvalue bounds on the pseudocodeword weight of expander codes," *Adv. in Math. of Comm.*, vol. 1, no. 3, pp. 287–307, Aug 2007.

[74] R. T. and R. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 638–656, Feb. 2001.

[75] S. Lin and D. J. Costello, *Error Control Coding, 2nd Edition*. Englewood Cliffs, NJ: Prentice-Hall, 2004.

[76] Y. Kou, S. Lin, and M. Fossorier, "Low-density parity-check codes based on finite geometries: A rediscovery and new results," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[77] X. Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2012.

[78] D. V. Nguyen, S. K. Chilappagari, M. W. Marcellin, and B. Vasic, "On the construction of structured LDPC codes free of small trapping sets," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2280–2302, Apr. 2012.

[79] D. MacKay, Online Database of Low-Density Parity-Check Codes. [Online]. Available: http://wol.ra.phy.cam.uk/mackay/codes/data.html.

[80] M. Davey and D. MacKay, "Low-density parity-check codes over GF (q)," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, June 1998.

[81] J. Colbourn and J. H. Dinitz, *The handbook of combinatorial designs*. Boca, Raton, FL, USA: CRC Press, 1996.

[82] T. Beth, D. Jungnickel, and H. Lenz, *Design theory*. New York, NY, USA: Cambridge University Press, 1986.

[83] R. Lucas, M. P. C. Fossorier, Y. Kou, and S. Lin, "Iterative decoding of one-step majority logic deductible codes based on belief propagation," *IEEE Transactions on Communications*, vol. 48, no. 6, pp. 931–937, June 2000.

[84] L. Lan, Y. Y. Tai, S. Lin, B. Memari, and B. Honary, "New constructions of quasi-cyclic LDPC codes based on special classes of BIBDs for the AWGN and the binary erasure channels," *IEEE Transactions on Communicatons*, vol. 56, no. 1, pp. 39–48, Jan. 2008.

[85] R. J. and P. Vontobel, "Construction of LDPC codes using Ramanujan graphs and ideas from Margulis," in *Proc. 38th Annu. Allerton Conf. Communication, Computing and Control*, Monticello, Il, USA, Oct. 2000, pp. 248–257.

[86] G. Margulis, "Explicit constructions of graphs without short cycles and low density codes," *Combinatorica*, vol. 2, no. 1, pp. 71–78, 1982.

[87] ——, "Explicit group-theoretic constructions for combinatorial designs with applications to expanders and concentrators," *Probl. Pered. Inform*, vol. 24, no. 1, p. 51–60, 1988.

[88] D. V. Nguyen and B. Vasic, "Two-Bit Bit Flipping Algorithms for LDPC Codes and Collective Error Correction," *IEEE Trans. Comm.*, vol. 62, no. 4, pp. 1153–1163, April 2014.

[89] X. Wu, C. Ling, M. Jiang, E. Xu, C. Zhao, and X. You, "New insights into weighted bit-flipping decoding," *IEEE Trans. Comm.*, vol. 57, no. 3, p. 591–596, Mar. 2009.

[90] A. Nouh and A. A. Banihashemi, "Reliability-based schedule for bit-flipping decoding of low-density parity-check codes," *IEEE Trans. Comm.*, vol. 52, no. 12, p. 2038–2040, Dec. 2004.

[91] T. Wadayama, K. Nakamura, M. Yagita, Y. Funahashi, S. Usami, and I. Takumi, "Gradient Descent Bit Flipping algorithms for decoding LDPC codes," *IEEE Trans. Comm.*, vol. 58, no. 6, pp. 1610–1614, June 2010.

[92] J. Chen, A. Dholakia, E. Eleftheriou, M. Fossorier, and X. Hu, "Reduced complexity decoding of ldpc codes," *IEEE Transactions on Communications*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.

[93] M. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low-density parity check codes based on belief propagation," *IEEE Transactions on Communications*, vol. 47, no. 5, p. 673–680, May 1999.

[94] S. Planjery, D. Declercq, L. Danjean, and B. Vasic, "Finite alphabet iterative decoders, part i: Decoding beyond belief propagation on the bsc," *IEEE Transactions on Communications*, vol. 61, no. 10, pp. 4033–4045, Oct. 2013.

[95] S. Planjery, D. Declercq, B. Vasic, and L. Danjean, "Finite alphabet iterative decoders for LDPC codes surpassing floating-point iterative decoders," *IET Electronics Letters*, vol. 46, no. 16, Aug. 2011.

[96] T. Richardson, "Error floors of LDPC codes," in *Proc. 41th Annu. Allerton Conf. Communication, Computing and Control*, Monticello, Il, USA, Oct. 2003.

[97] M. Ivkovic, S. Chilappagari, and B. Vasic, "Eliminating trapping sets in low-density parity-check codes by using Tanner graph covers," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, p. 3763–3768, 2008.

[98] M. Karimi and A. Banihashemi, "An efficient algorithm for finding dominant trapping sets of irregular LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, St. Petersburg, Russia, July-Aug. 2011, pp. 1091–1095.

[99] S. Laendner and O. Milenkovic, "Algorithmic and combinatorial analysis of trapping sets in structured LDPC codes," in *Proc. IEEE International Conference on Wireless Networks, Communications and Mobile Computing*, Hawaii, USA, June 2005, pp. 630–635.

[100] A. McGregor and O. Milenkovic, "On the hardness of approximating stopping and trapping sets in LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, p. 1640–1650, Apr. 2010.

[101] S. K. Chilappagari and B. Vasic, "Error-correction capability of column-weight-three LDPC codes," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2055–2061, May 2009.

[102] S. K. Chilappagari, D. V. Nguyen, B. Vasic, and M. W. Marcellin, "Error correction capability of column-weight-three LDPC codes under the Gallager A algorithm - Part II," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2626–2639, June 2010.

[103] B. Vasic, S. Chilappagari, D. Nguyen, and S. Planjery, "Trapping set ontology," in *Proc. 47th Annu. Allerton Conf. Communication, Computing and Control*, Monticello, Il, USA, Sep. 2009, pp. 1–7.

[104] S. K. Chilappagari, D. V. Nguyen, B. Vasic, and M. W. Marcellin, "On trapping sets and guaranteed error correction capability of LDPC codes and GLDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1600–1611, Apr. 2010.

[105] C. Di, D. Proietti, I. Telatar, T. Richardson, and L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, June 2002.

[106] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright, "LP decoding corrects a constant fraction of errors," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, p. 82–89, Jan. 2007.

[107] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson, "Randomness conductors and constant-degree lossless expanders," in *STOC '02: Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, New York, NY, USA: ACM Press, 2002, pp. 659–668.

[108] X. Wei and A. Akansu, "Density evolution forlow-density parity-check codes under max-log-MAP decoding," *IET Electronic Letters*, vol. 37, no. 18, pp. 1125–1126, Aug. 2001.

[109] O. Al Rasheed, P. Ivanis, and B. Vasic, "Fault-tolerant probabilistic gradient-descent bit flipping decoder," *IEEE Communications Letters*, vol. 18, no. 9, pp. 1487–1490, Sept. 2014.

[110] P. Ivanis, O. Al Rasheed, and B. Vasic, "MUDRI: A fault-tolerant decoding algorithm," in *in IEEE Int. Conf. on Commun. (ICC 2015)*, London, June 2015, pp. 4291–4296.

[111] T. Ngatched, M. Bossert, A. Fahrner, and F. Takawira, "Two bit-flipping decoding algorithms for low-density parity-check codes," *IEEE Trans. Comm.*, vol. 57, no. 3, p. 591–596, Mar. 2009.

[112] G. Sundararajan, C. Winstead, and E. Boutillon, "Noisy gradient descent bit-flip decoding for LDPC codes," *IEEE Trans. Comm.*, vol. 62, no. 10, pp. 3385–3400, Oct. 2014.

[113] L. D. Rudolph, "A class of majority logic decodable codes," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 305–307, Apr. 1967.

[114] J. L. Massey, *Threshold Decoding*. Cambridge, MA, USA: MIT Press, 1963.

[115] R. Radhakrishnan, S. Sankaranarayanan, and B. Vasic, "Analytical performance of one-step majority logic decoding of regular LDPC codes," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 2007)*, Nice, France, June 2007, pp. 231–235.

[116] S. Chilappagari, M. Ivkovic, and B. Vasic, "Analysis of one step majority logic decoders constructed from faulty gates," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 2006)*, Seattle, USA, July 2006, pp. 469–473.

[117] S. Brkic, P. Ivanis, and B. Vasic, "Analysis of one-step majority logic decoding under correlated data-dependent gate failures," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 2014)*, Honolulu, USA, June-July 2014, pp. 2599–2603.

[118] S. Brkic, , P. Ivanis, and B. Vasic, "Guaranteed error correction of faulty bit-flipping decoders under data-dependent gate failures," in *Proc. of IEEE International Symposium on Information Theory (ISIT 2016)*, Barcelona, Spain, July 2016.

[119] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Annals of Mathematical Statistics*, vol. 23, p. 493–507, 1952.

[120] N. Alon, S. Hoory, and M. Linial, "The moore bound for irregular graphs," *Graphs and Combinatorics*, vol. 18, no. 1, p. 53–57, 2002.

[121] M. Mihaljevic and J. Golic, "A method for convergence analysis of iterative probabilistic decoding," *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 2206–2211, Sep. 2000.

[122] C. H. Huang, Y. Li, and L. Dolecek, "Gallager B LDPC decoder with transient and permanent errors," *IEEE Transactions on Communications*, vol. 62, no. 1, pp. 15–28, Jan. 2014.

[123] F. Leduc-Primeau and W. Gross, "Faulty Gallager-B decoding with optimal message repetition," in *Proceedings of 50th Allerton Conference on Communication, Control, and Computing*, Monticello, USA, Oct. 2012, pp. 549–556.

[124] C. Ngassa, V. Savin, E. Dupraz, and D. Declercq, "Density evolution and functional threshold for the noisy min-sum decoder," *IEEE Trans. Commun.*, vol. 63, no. 5, pp. 1497–1509, May 2015.

[125] C. Ngassa, V. Savin, and D. Declercq, "Unconventional behavior of the noisy min-sum decoder over the binary symmetric channel," in *Proc. Information Theory and Applications Workshop*, Feb. 2014, p. 1–10.

[126] I. Perez-Andrade, X. Zuo, R. Maunder, B. Al-Hashimi, and L. Hanzo, "Analysis of voltage- and clock-scaling-induced timing errors in stochastic LDPC decoders," in *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, Shanghai, Apr. 2013, p. 4293–4298.

[127] S. Brkic, O.-A. Rasheed, P. Ivanis, and V. B., "On fault-tolerance of the gallager b decoder under data-dependent gate failures," *IEEE Communications Letters*, vol. 19, no. 8, p. 1299–1302, Avg. 2015.

[128] B. Vasic, P. Ivanis, S. Brkic, and R. V., "Fault-resilient decoders and memories made of unreliable components," in *Proceedings of 10th Information Theory and Applications Workshop (ITA 2015)*, San Diego, CA, Feb. 2015, paper 273, [Online Available:] http://ita.ucsd.edu/workshop/15/files/paper/paper 273.pdf.

[129] E. Janulewicz and A. Banihashemi, "Performance analysis of iterative decoding algorithms with memory over memoryless channels," *IEEE Trans. Commun.*, vol. 16, no. 12, pp. 3556–3566, Dec. 2014.

[130] N. Mobini, A. Banihashemi, and S. Hemati, "A differential binary message-passing LDPC decoder," *IEEE Trans. Comm.*, vol. 57, no. 9, pp. 2518–2523, Sep. 2009.

[131] L. Sassatelli, S. Chilappagari, B. Vasic, and D. Declercq, "Two-bit message passing decoders for ldpc codes over the binary symmetric channel," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 2009)*, Seoul, Korea, June-July 2009, pp. 2156–2160.

[132] S. Brkic, , B. Vasic, P. Ivanis, and D. D., "Message-aggregation-enhanced iterative hard-decision decoders," in *Proc. Information Theory and Applications Workshop (ITA)*, San Diego, USA, Jan.-Feb. 2016.

[133] S. Planjery, D. Declercq, M. Diouf, and B. Vasic, "On the guaranteed error-correction of decimation-enhanced iterative decoders," in *8th International Symposioum on Turbo Codes and Iterative Information Processing (ISTC)*, Bremen, Germany, Aug. 2014, pp. 57–61.

[134] D. Declercq, *List of LDPC codes*, http://www2.engr.arizona.edu/ vasiclab/tool.php?id=7.

[135] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Trans. Device Mater. Reliabil.*, vol. 5, no. 3, pp. 301–316, Sep. 2005.

[136] C. W. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," *IEEE Trans. Device Mater. Reliabil.*, vol. 5, no. 3, p. 397–404, Sep. 2005.

[137] D. Spielman, "Highly fault-tolerant parallel computation," in *Proc. IEEE Conference on Foundations of Computer Science*, 1996, pp. 154–163.

[138] S. K. Chilappagari and B. Vasic, "Reliable memories built from unreliable components based on expander graphs," *arXiv:0705.0044v1 [cs.IT]*, May 2007.

[139] E. Dupraz, Declercq, and B. Vasic, "Analysis of Taylor-Kuznetsov memory using one-step majority logic decoder," in *Proceedings of 10th Information Theory and Applications Workshop (ITA 2015)*, San Diego, CA, Feb. 2015, paper 273, [Online Available:] http://ita.ucsd.edu/workshop/15/files/paper/paper 3446.pdf.

[140] M. Ivkovic, S. K. Chilappagari, and B. Vasic, "Construction of memory circuits using unreliable components based on low-density parity-check codes," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '06)*, San Francisco, CA, USA, Nov. 2006, pp. 1–5.

[141] A. Khajeh, K. Amiri, M. Khairy, A. M. Eltawil, and F. Kurdahi, "A unified hardware and channel noise model for communication systems," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '10)*, Miami, Florida, USA, 6-10 Dec. 2010, pp. 1–5.

[142] S. Brkic, P. Ivanis, and B. Vasic, "Reliability of memories built from unreliable components under data-dependent gate failures," *IEEE Communiaction Letters*, vol. 19, no. 12, pp. 2098–2101, Dec. 2015.

[143] B. Vasic, P. Ivanis, and S. Brkic, "Low complexity memory architectures based on LDPC codes: Benefits and disadvantages," in *Proc. 12th International Conference on Telecommunication in Modern Satellite, Cable and Broadcasting Services (TELSIKS 2015)*, 14-17 Oct. 2015, pp. 11–18.

[144] S. Schechter, G. H. Lohy, K. Strauss, and D. Burger, "Use ECP, not ECC, for hard failures in resistive memories," in *Proc. International Symposium on Computer Architecture - ISCA*, June 2010.

[145] C. Heegard, "Partitioned linear block codes for computer memory with 'stuck-at' defects," *IEEE Transactions on Inforamtion Theory*, vol. 29, no. 6, pp. 831–842, Nov. 1983.

[146] Y. Kim and V. K. V. Kumar, "Coding for memory with stuck-at defects," *[Online Available] http://arxiv.org/ftp/arxiv/papers/ 1304/1304.4821.pdf*.

[147] N. Mielke, T. Marquart, N. Wu, J. Kessenich, H. Belgal, E. Schares, F. Trivedi, E. Goodness, and L. Nevill, "Bit error rate in NAND flash memories," in *Proc. IEEE Int. Rel. Phys. Symp.*, Jeju Island, Korea, 2008, pp. 9–19.

[148] P. Ankolekar, R. Isaac, and J. Bredow, "Multibit error-correction methods for latency-constrained flash memory systems," *IEEE Transactions on Device and Materials Reliability*, vol. 10, no. 1, pp. 2098–2101, Mar. 2010.

[149] C. Chr, S. Su, and S. Wu, "New step-by-step decoding for binary BCH codes," in *Proc. 9th Int. Conf. Commun. Syst.*, Sep. 2004, p. 456–460.

[150] F. Sun, S. Devarajan, K. Rose, and T. Zhang, "Multilevel flash memory on-chip error correction based on trellis coded modulation," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2006, p. 1443–1446.

[151] F. Zhang, H. Pfister, and A. Jiang, "Multilevel flash memory on-chip error correction based on trellis coded modulation," in *Proc. IEEE International Symposium on Information Theory (ISIT 2010)*, Austin, Texas, U.S.A., June 13 - 18 2010, p. 859–863.

[152] L. S., P. Reviriego, and J. Maestro, "Efficient majority logic fault detection with difference-set codes for memory applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 1, pp. 148–156, Jan. 2012.

[153] J. Wang, K. Vakilinia, T. Chen, T. Courtade, G. Dong, T. Zhang, H. Shankar, and R. Wesel, "Enhanced precision through multiple reads for LDPC decoding in flash memories," *IEEE Journal Sel. Areas Commun.*, vol. 32, no. 5, pp. 880–891, May 2014.

[154] K. Haymaker and C. Kelley, "Geometric WOM codes and coding strategies for multilevel flash memories," *Designs, Codes and Cryptography*, vol. 70, no. 1, pp. 91–104, Jan. 2014.

[155] E. En Gad, W. Huang, L. Yue, and J. Bruck, "Rewriting flash memories by message passing," in *Proc. IEEE International Symposium on Information Theory (ISIT 2015)*, Hong Kong, 14-19 June 2015, pp. 646–650.

[156] M. Lefter, G. Voicu, and S. Cotofana, "A shared polyhedral cache for 3d wide-i/o multicore computing platforms," in *Proc. IEEE International Symposium in Circuits and Systems (ISCAS)*, Hong Kong, May 2015, pp. 425–428.

[157] L. Varshney, "Toward limits of constructing reliable memories from unreliable components," in *Proc. IEEE Information Theory Workshop (ITW)*, Jeju Island, Korea, 11-15 Oct. 2015.

Прилог 1.

Изјава о ауторству

Потписани-а СРЂАН ЂРКИЋ

број уписа 5004/2010

Изјављујем

да је докторска дисертација под насловом

ДЕКОДОВАЊЕ КОДОВА СА МАЛОМ ГУСТИНОМ ПРОВЕРА ПАРНОСТИ У ПРИСУСТВУ ГРЕШАКА У ЛОГИЧКИМ КОЛИМА

- резултат сопственог истраживачког рада,
- да предложена дисертација у целини ни у деловима није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио/ла ауторска права и користио интелектуалну својину других лица.

Потпис докторанда

У Београду, 23.06.2016.

Срђан Ђрић

Прилог 2.

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора СРЂАН ВРКИЋ

Број уписа 5004/2010

Студијски програм ТЕЛЕКОМУНИКАЦИЈЕ

Наслов рада ДЕКОДОВАЊЕ КОДОВА СА МАЛОМ ГУСТИНОМ ПРОВЕРА ПАРНОСТИ У ПРИСУСТВУ ПРЕВАКА У ЛОГИЧКИМ КОЛИМА

Ментор ПРЕДРАГ ИВАНИШ

Потписани СРЂАН ВРКИЋ

изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао/ла за објављивање на порталу **Дигиталног репозиторијума Универзитета у Београду.**

Дозвољавам да се објаве моји лични подаци везани за добијање академског звања доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

Потпис докторанда

У Београду, 23.06.2016

Срђан Вркић

Прилог 3.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

ДЕКОДОВАЊЕ КОДОВА СА МАЛОМ ГУСТИНОМ ПРОВЕРА ПАРНОСТИ У ПРИСУСТВУ ГРЕШАКА У ЛОГИЧКИМ КОЛИМА

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитални репозиторијум Универзитета у Београду могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио/ла.

1. Ауторство

(2) Ауторство - некомерцијално

3. Ауторство – некомерцијално – без прераде

4. Ауторство – некомерцијално – делити под истим условима

5. Ауторство – без прераде

6. Ауторство – делити под истим условима

(Молимо да заокружите само једну од шест понуђених лиценци, кратак опис лиценци дат је на полеђини листа).

Потпис докторанда

У Београду, 23.06.2016.

Срђан Вукић

1. Ауторство - Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.

2. Ауторство – некомерцијално. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.

3. Ауторство - некомерцијално – без прераде. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.

4. Ауторство - некомерцијално – делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.

5. Ауторство – без прераде. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.

6. Ауторство - делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.