A SEGMENTATION-BASED BISR SCHEME

Georgios Zervakis, Nikolaos Eftaxiopoulos, Kostas Tsoumanis, Nicholas Axelos, Kiamal Pekmestzi

NATIONAL TECHNICAL UNIVERSITY OF ATHENS

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING
DIVISION OF COMPUTER SCIENCE

Introduction (1/3)

- SRAM memory: Determining factor in yield loss and reliability problems in modern SoC (System on Chip)
- Small geometries, Voltage scaling → Manufacturing defects and process variation
- Need for fault tolerance techniques

Introduction (2/3)

- Conventional techniques
  - Use of spare rows and columns
  - External test equipment creates fault map
  - Faulty elements are replaced by spare ones
  - Costly and time consuming
- BISR (Built-in Self-Repair) techniques
  - Move repair process on-chip
  - Repairing circuitry intervening in the decoding operation
  - Localization of faulty cells by a BIST (Built-In Self-Test) circuitry

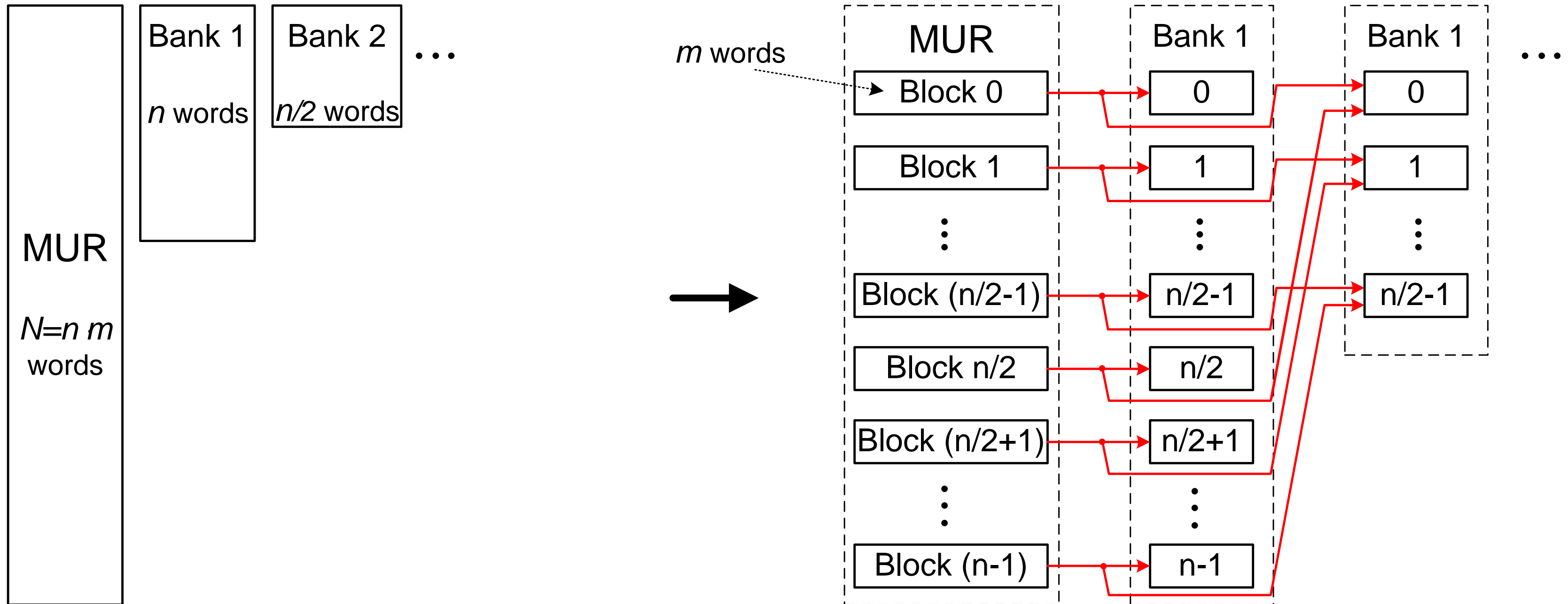
Introduction (3/3)

- Granularity of repair affects remapping circuitry overhead and utilization of spare elements
  - Row/Column-based techniques → Simple circuitry, Not efficient allocation of spare resources
  - Block/bit-based techniques → Complex circuitry, Efficient allocation of spare resources
- Considering the tradeoff, a cache-based architecture with multiple banks was proposed

Previous work (1/2)

- Multiple cache banks that repair faulty elements in word or bit level
  - Faulty words are directly mapped to redundant cache banks.
  - Each cache bank has half the lines of its previous one.
  - Mathematical analysis proves that for λ faulty words in the MUR (Memory under repair), ~100% reparability can be achieved using this technique if the primary cache bank has $n=2\lambda$ lines.
  - The bit level technique reduces the area overhead comparing to the word level one.

Previous work (2/2)



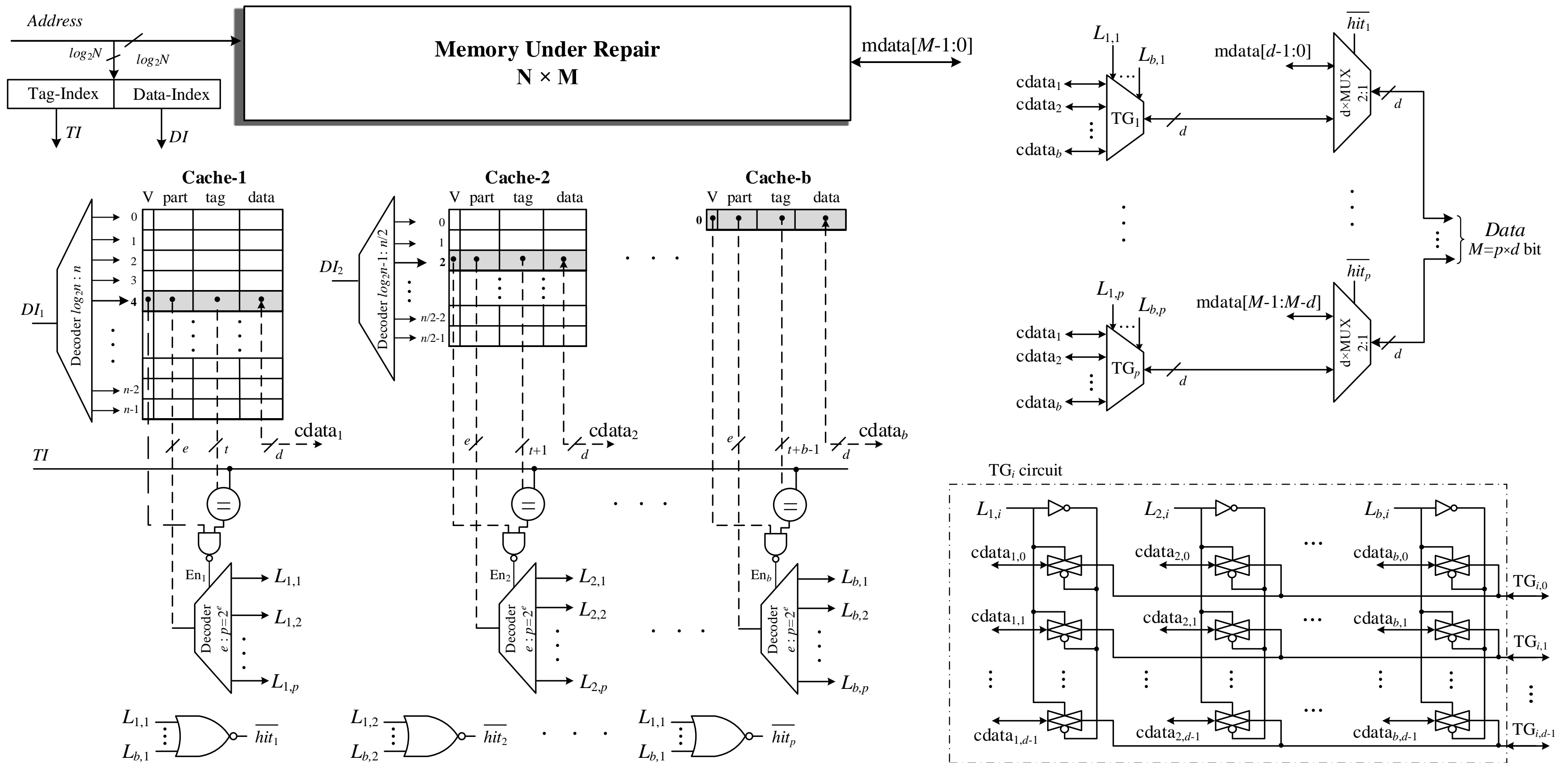
Proposed BISR scheme (1/6)

- A segmentation-based BISR scheme is proposed
  - The MUR (Memory Under Repair) word is segmented in parts and can be repaired in any segmentation level
  - A part of the *address* is used as an *index* (DI) to map the repaired word in the cache banks and another part is used as a *tag* (TI) to identify the exact word that is mapped
  - A special field (*part*) in each cache line identifies which segment of the MUR word is repaired

Proposed BISR scheme (2/6)

- Except for the cache banks there is a remapping and routing circuit
  - Identifies if parts of the requested word are stored in the cache bank circuitry and drives them to the data bus.
- Interaction with BIST
  - A BIST circuitry has previously identified all the faulty parts of the MUR and distributed them in the spare cache banks.

Proposed BISR scheme (3/6)

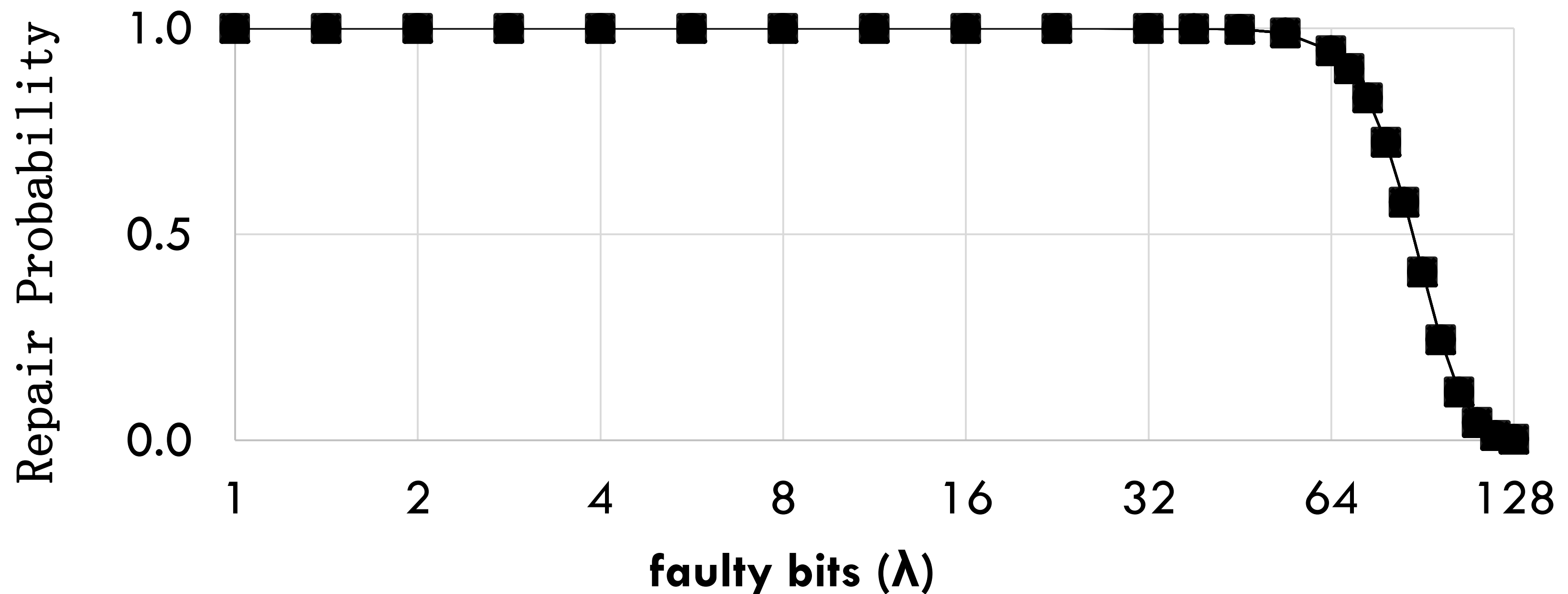


Proposed BISR scheme (4/6)

- A mathematical analysis proved that:
  - The probability of the MUR being fully repaired by the BISR circuitry does not depend on the level of segmentation.
  - For λ faulty bits in the MUR, a primary bank with $n=2\lambda$ lines is required to achieve ~100% reparability independently of the segmentation level.
  - Negligible differences in the probability of repair are observed only for values of faulty bits ($\lambda>2n$) that even the non-segmented scheme cannot repair with ~100% probability.

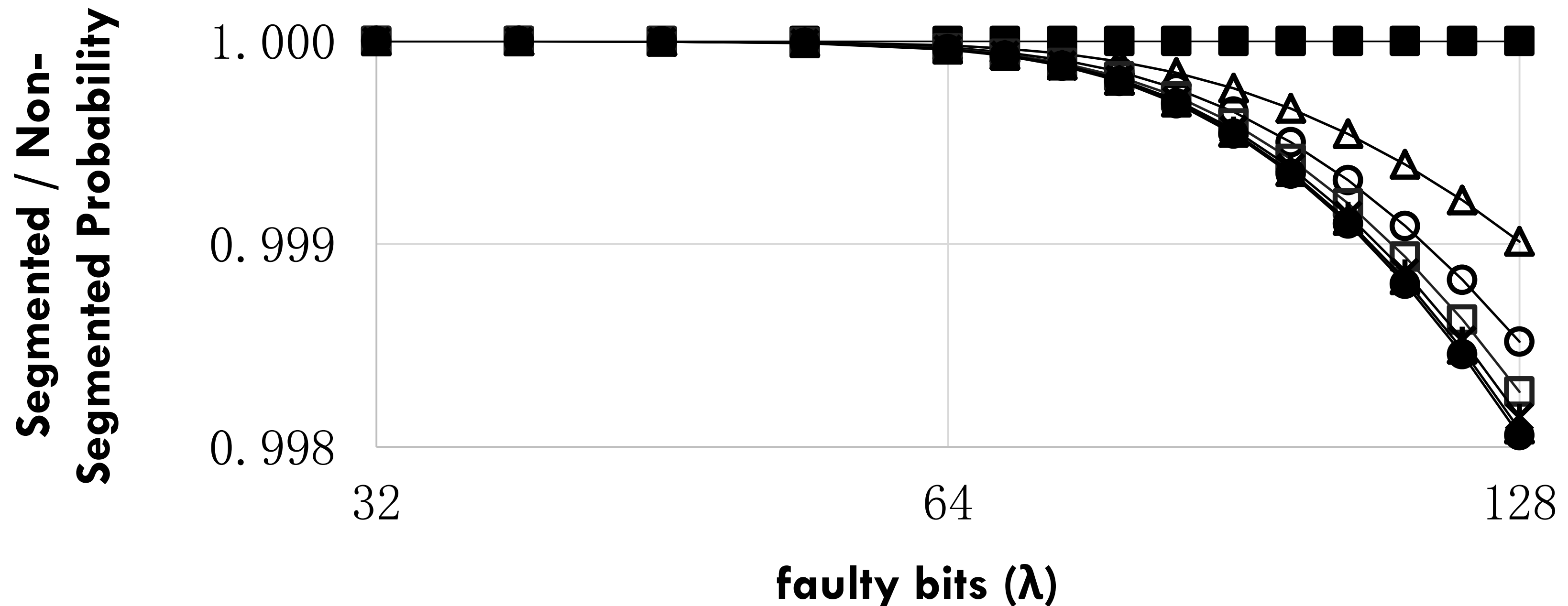
Proposed BISR scheme (5/6)

- MUR 1Mx64, primary cache bank with 64 lines



Proposed BISR scheme (6/6)

- MUR 1Mx64, primary cache bank with 64 lines



Exploration of optimal segmentation (1/7)

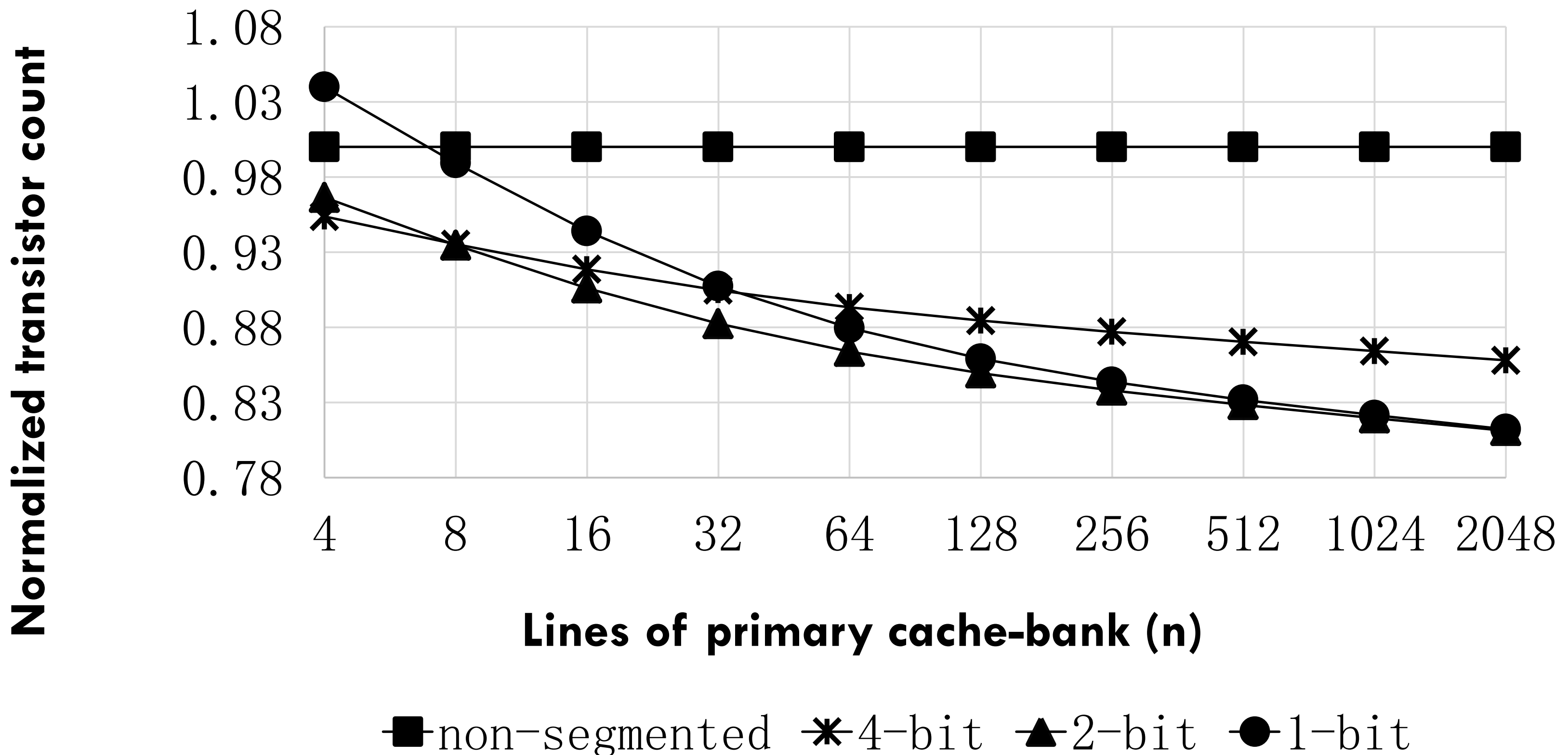
- The level of segmentation does not affect the ~100% reparability.
- The basic criteria to find the optimal segmentation is the area occupied (transistor count) by the BISR circuitry.
  - Segmenting in bit-level seems an obvious solution but the additional remapping and routing circuitry becomes significantly larger.
  - The area occupied by the BISR circuitry for each level of segmentation was computed using a unit gate model.

Exploration of optimal segmentation (2/7)

- The area of the BISR circuitry is affected by the:
  - Size of the MUR (*N* lines, *M*-bit words)
    - An exhaustive exploration about the effect of the MUR lines (*N*) was conducted and the impact proved to be insignificant → The number of MUR lines is set to $N=2^{20}$
    - The most usual memory word lengths (8, 16, 32 and 64-bit) are used in the exploration
  - Lines of primary cache bank (*n*)
    - The exploration was conducted for a range of lines of the primary cache bank from 4 to 2048
- All the possible segmentations were explored

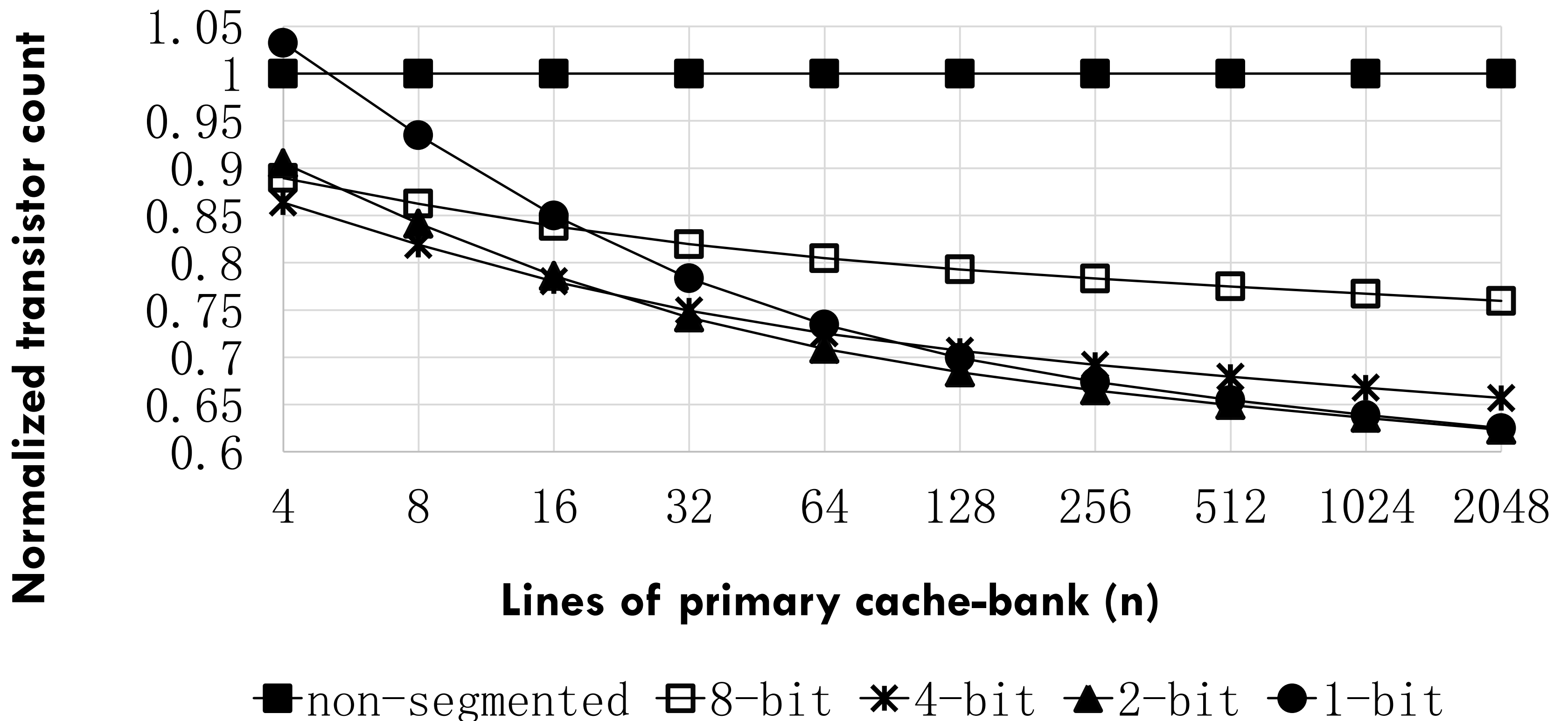
Exploration of optimal segmentation (3/7)

- MUR word length = 8 bits



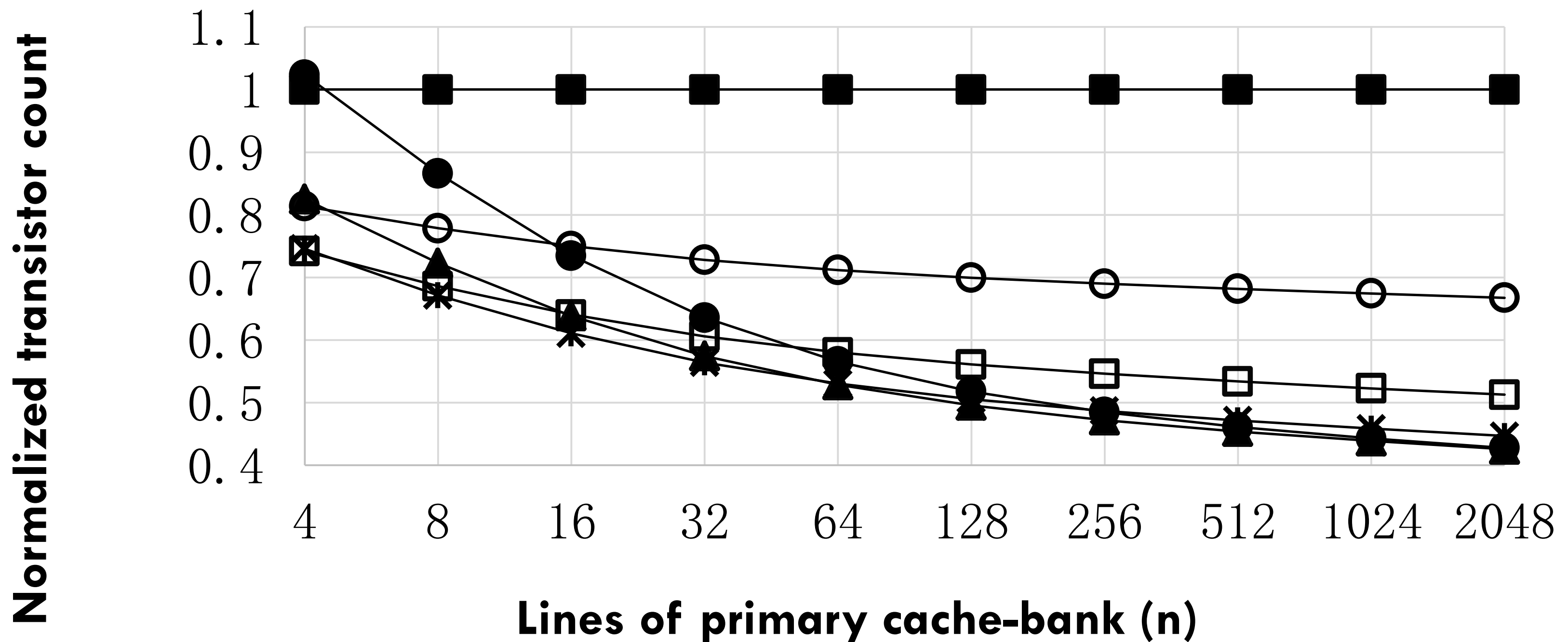
Exploration of optimal segmentation (4/7)

- MUR word length = 16 bits



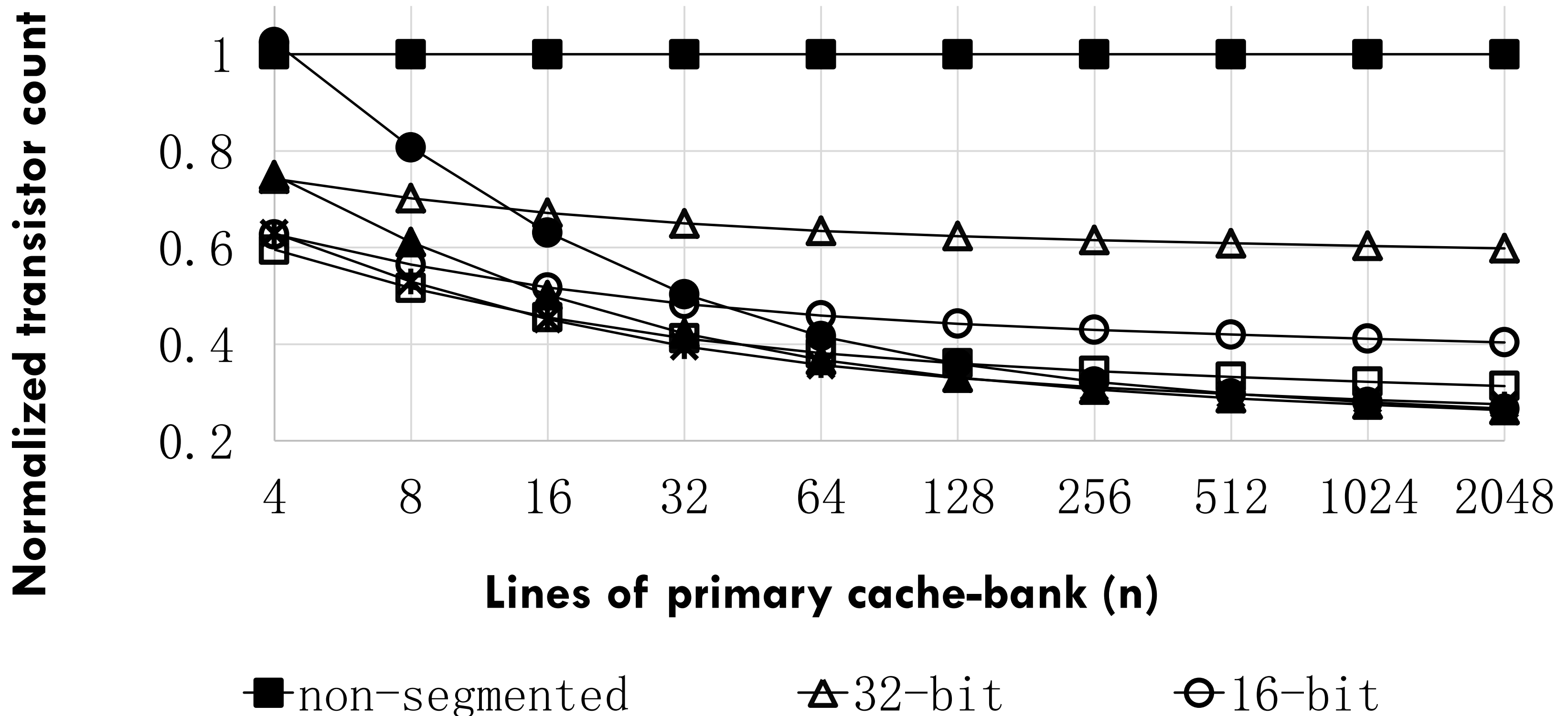
Exploration of optimal segmentation (5/7)

- MUR word length = 32 bits



Exploration of optimal segmentation (6/7)

- MUR word length = 64 bits



Exploration of optimal segmentation (7/7)

- Overall observation of the results
  - The non-segmented scheme is the one with the highest area overhead and it must be avoided.
  - The 1-bit segmentation, although it might seem as the obvious solution, never proves to be the best option.
  - When the MUR has a high fault rate, the 2-bit segmentation seems to be the optimal choice.
  - When the MUR fault rate becomes smaller and the word length increases, segmentation in bigger groups (4-bit, 8-bit) should be preferred.

Conclusion

- A BISR scheme with parametric repair level is proposed
- The proposed scheme can operate for all the possible segmentations of the MUR word
- According to a mathematical analysis the reparability is not affected by the segmentation level
- In terms of transistor count the non-segmented scheme proves to have the highest area overhead while the 2-bit segmentation proves to be the optimal for high fault rates of the MUR

Acknowledgements

- This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thales -UOA- HOLISTIC. Investing in knowledge society through the European Social Fund.

Thank You!

- Questions?