Electronic Colloquium on Computational Complexity, Report No. 68 (2026)

Exponential-Size Circuit Complexity is Comeager in Symmetric Exponential Time

John M. Hitchcock*

Abstract

Lutz (1987) introduced resource-bounded category and showed the circuit size class $\mathsf{SIZE}(\frac{2^n}{n})$ is meager within ESPACE. Li (2024) established that the symmetric alternation class $\mathsf{S}_2^{\mathsf{E}}$ contains problems requiring circuits of size $\frac{2^n}{n}$.

In this note, we extend resource-bounded category to $\mathsf{S}_2^{\mathsf{E}}$ by defining meagerness relative to single-valued $\mathsf{FS}_2^{\mathsf{P}}$ strategies in the Banach-Mazur game. We show that Li's $\mathsf{FS}_2^{\mathsf{P}}$ algorithm for the Range Avoidance problem yields a winning strategy, proving that $\mathsf{SIZE}(\frac{2^n}{n})$ is meager in $\mathsf{S}_2^{\mathsf{E}}$.

Consequently, languages requiring exponential-size circuits are comeager in $\mathsf{S}_2^{\mathsf{E}}$: they are typical with respect to resource-bounded category.

1 Introduction

Resource-bounded category, introduced by Lutz [14–16], provides a way to analyze the typical properties of complexity classes using topological notions of size. Lutz showed that the class $\mathsf{SIZE}(\frac{2^n}{n})$ of languages decidable by circuits of size $\frac{2^n}{n}$ is *meager* within the exponential-space class ESPACE.

Theorem 1.1. (Lutz [14–16]) $\mathsf{SIZE}(\frac{2^n}{n})$ *is meager in* ESPACE.

The symmetric exponential-time class $\mathsf{S}_2^{\mathsf{E}}$ is an exponential version of the class $\mathsf{S}_2^{\mathsf{P}}$ [5, 21] and it lies between E and ESPACE. Building on work by Korten [12] and Chen, Hirahara, and Ren [6], Li [13] recently proved that $\mathsf{S}_2^{\mathsf{E}}$ contains languages that require near-maximum circuit size.

Theorem 1.2. (Li [13]) $\mathsf{S}_2^{\mathsf{E}} \not\subseteq \mathsf{SIZE}(\frac{2^n}{n})$.

This solved a longstanding open problem to show the second level of the exponential-time hierarchy requires exponential-size circuits [1–3, 9, 10, 19, 20]. Li's proof technique involves constructing a hard language using a *single-valued* $\mathsf{FS}_2^{\mathsf{P}}$ algorithm (a function problem analogue of $\mathsf{S}_2^{\mathsf{P}}$). Given these developments, a natural question is whether the category framework can be refined to operate within $\mathsf{S}_2^{\mathsf{E}}$. Can we show that languages requiring large circuits are not just present, but are in fact *typical* within $\mathsf{S}_2^{\mathsf{E}}$?

In this note, we answer this question affirmatively. We adapt the resource-bounded category machinery to the class $\mathsf{S}_2^{\mathsf{E}}$ by defining meagerness relative to single-valued $\mathsf{FS}_2^{\mathsf{P}}$-computable constructors in the Banach-Mazur game. Our main result shows that, analogous to Lutz's result for ESPACE, small circuits are atypical within $\mathsf{S}_2^{\mathsf{E}}$.

Theorem 1.3. $\mathsf{SIZE}(\frac{2^n}{n})$ *is meager in* $\mathsf{S}_2^{\mathsf{E}}$.

We prove this by demonstrating that Li's single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm for the range avoidance problem [6, 12] provides a winning strategy against $\mathsf{SIZE}(\frac{2^n}{n})$ in the appropriate Banach-Mazur game. Thus, languages requiring exponential-size circuits are not just present in $\mathsf{S}_2^{\mathsf{E}}$—they are typical in the sense of resource-bounded category: the class of such languages is comeager in $\mathsf{S}_2^{\mathsf{E}}$.

This paper is organized as follows. We present preliminaries on resource-bounded category and symmetric alternation in Section 2. Our results are in Section 3. We conclude in Section 4 with an open question about extending our results to resource-bounded measure.

*Department of Electrical Engineering and Computer Science, University of Wyoming. jhitchco@uwyo.edu. This research was supported in part by NSF grant 2431657.

ISSN 1433-8092

2 Preliminaries

2.1 Resource-Bounded Category

Resource-bounded category was introduced by Lutz [14–16]. The *Cantor space* C is the set of all infinite binary sequences. A *language* (or *decision problem*) is a subset of $\{0,1\}^*$. We identify each language with the element of Cantor space that is its characteristic sequence according to the standard enumeration $s_0 = \lambda, s_1 = 0, s_2 = 1, s_3 = 00, \ldots$ of $\{0,1\}^*$. In this way, complexity classes (sets of languages) are viewed as subsets of Cantor space.

Baire category classifies sets into two types: *first category* and *second category*. First category sets are also commonly called *meager*. A set is meager if it is a countable union of nowhere dense sets. An equivalent definition comes from Banach-Mazur games using functions called constructors.

Definition 2.1. A *constructor* is a total, single-valued function $\delta : \{0,1\}^* \to \{0,1\}^*$ which satisfies $x \underset{\neq}{\sqsubset} \delta(x)$ for all $x \in \{0,1\}^*$. The *result* of a constructor is the unique sequence $R(\delta) \in \mathsf{C}$ that extends $\delta^{(n)}(\lambda)$ for all n.

Let $X \subseteq \mathsf{C}$ and let Γ_{I} and Γ_{II} be two classes of functions. In the *Banach-Mazur game* $G[X; \Gamma_{\mathrm{I}}, \Gamma_{\mathrm{II}}]$ there are two players I and II. A *strategy* in the game is a constructor. In a play of the game, player I chooses a strategy $\gamma \in \Gamma_{\mathrm{I}}$ and player II chooses a strategy $\delta \in \Gamma_{\mathrm{II}}$. The *result* of this play is the sequence $R(\gamma, \delta) = R(\delta \circ \gamma)$. Intuitively, the result is the sequence obtained when the two players start with the empty string and take turns extending it with their strategies. A *winning strategy* for player II is a strategy $\delta \in \Gamma_{\mathrm{II}}$ such that for every $\gamma \in \Gamma_{\mathrm{I}}$, $R(\gamma, \delta) \notin X$.

Theorem 2.2. (Banach and Mazur) *A class $X \subseteq \mathsf{C}$ is meager if and only if player II has a winning strategy in the game $G[X; \mathsf{all}, \mathsf{all}]$.*

In resource-bounded category, Δ denotes a *resource bound* [16, 17]. Examples of Δ include:

$$\begin{aligned} \mathsf{all} &= \{f \mid f : \{0,1\}^* \to \{0,1\}^*\} \\ \mathsf{p} &= \{f \mid f \text{ is polynomial-time computable}\} \\ \mathsf{pspace} &= \{f \mid f \text{ is polynomial-space computable}\} \end{aligned}$$

For a resource bound Δ, we define the *result class*

$$R(\Delta) = \{R(\delta) \mid \delta \in \Delta \text{ is a constructor}\}.$$

Then $R(\mathsf{all}) = \mathsf{C}$, $R(\mathsf{p}) = \mathsf{E}$, and $R(\mathsf{pspace}) = \mathsf{ESPACE}$ [16]. Resource-bounded category [16] is defined by requiring player II's winning strategy to be computable within a resource bound Δ.

Definition 2.3. Let $X \subseteq \mathsf{C}$ and let Δ be a resource bound.

1. X is Δ-*meager* if player II has a winning strategy in the game $G[X; \mathsf{all}, \Delta]$.
2. X is Δ-*comeager* if X^c is Δ-meager.
3. X is *meager in* $R(\Delta)$ if $X \cap R(\Delta)$ is Δ-meager.
4. X is *comeager in* $R(\Delta)$ if X^c is meager in $R(\Delta)$.

The *resource-bounded Baire category theorem* [16] tells us that $R(\Delta)$ is not Δ-meager.

2.2 Symmetric Alternation

Symmetric alternation is a concept in computational complexity theory introduced independently by Canetti [5] and Russell and Sundaram [21] with different, but equivalent definitions [4]. We follow the definition due to Canetti.

Definition 2.4. Let $T\colon \mathbb{N} \to \mathbb{N}$. We say that a language $L \in \mathsf{S}_2\mathsf{TIME}[T(n)]$ if there exists an $O(T(n))$-time verifier $V(x, \pi_1, \pi_2)$ that takes $x \in \{0,1\}^n$ and $\pi_1, \pi_2 \in \{0,1\}^{T(n)}$ as input, satisfying

$$(\exists \pi_1)(\forall \pi_2) V(x, \pi_1, \pi_2) = \chi_L(x)$$

and

$$(\exists \pi_2)(\forall \pi_1) V(x, \pi_1, \pi_2) = \chi_L(x).$$

Definition 2.5. We define the symmetric alternation complexity classes

$$\mathsf{S}_2^{\mathsf{P}} = \bigcup_{c=1}^{\infty} \mathsf{S}_2\mathsf{TIME}[n^c].$$

and

$$\mathsf{S}_2^{\mathsf{E}} = \bigcup_{c=1}^{\infty} \mathsf{S}_2\mathsf{TIME}[2^{cn}].$$

Symmetric alternation is used to compute single-valued functions as follows [13].

Definition 2.6. A *single-valued* $\mathsf{FS}_2^{\mathsf{P}}$ algorithm A is specified by a polynomial $\ell(\cdot)$ together with a polynomial-time verifier $V_A(x, \pi_1, \pi_2)$. On input $x \in \{0,1\}^*$, we say that A *outputs* a string $y_x \in \{0,1\}^*$ if the following two conditions hold:

1. There exists $\pi_1 \in \{0,1\}^{\ell(|x|)}$ such that for every $\pi_2 \in \{0,1\}^{\ell(|x|)}$,

$$V_A(x, \pi_1, \pi_2) \;=\; y_x.$$

2. There exists $\pi_2 \in \{0,1\}^{\ell(|x|)}$ such that for every $\pi_1 \in \{0,1\}^{\ell(|x|)}$,

$$V_A(x, \pi_1, \pi_2) \;=\; y_x.$$

Note that Definition 2.4 prescribes a single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm for χ_L.

Li [13] used a single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm for the range avoidance problem [6, 12] to prove Theorem 1.2 that $\mathsf{S}_2^{\mathsf{E}}$ requires exponential-size circuits. The following is a corollary to Li's proof.

Theorem 2.7. (Li [13]) *There is an* $\mathsf{FS}_2^{\mathsf{P}}$ *algorithm* $\mathcal{A}$ *that on input* 0^{2^n} *outputs a truth-table* $f \in \{0,1\}^{2^n}$ *such that* f *has circuit complexity at least* $\frac{2^n}{n}$.

Remark 2.8. The $\frac{2^n}{n}$ bound in Theorem 1.2, Theorem 2.7, and our results in Section 3 may be replaced by $\frac{2^n}{n}\left(1 + \frac{\alpha \log n}{n}\right)$ for any fixed $\alpha \in (0,1)$ [7, 8, 17]. We use $\frac{2^n}{n}$ to simplify notation.

3 Category in Symmetric Exponential Time

Recall that $R(\mathsf{p}) = \mathsf{E}$ and $R(\mathsf{pspace}) = \mathsf{ESPACE}$. We now show that the result class of $\mathsf{FS}_2^{\mathsf{P}}$ is $\mathsf{S}_2^{\mathsf{E}}$.

Lemma 3.1. $R(\mathsf{FS}_2^{\mathsf{P}}) = \mathsf{S}_2^{\mathsf{E}}$.

Proof. Let $L = R(A)$ where A is a single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm. Let V_A be as in Definition 2.6. On input x, let n be the index such that $s_n = x$. The idea is to start with $z_0 = \lambda$ and compute $z_{i+1} = A(z_i)$ until $|z_i| > n$. Then the answer for x is $z_i[n]$. We make at most $n \leq 2^{|x|}$ calls to A. Formally, define a verifier V that on input $s_n = x$ also takes two tuples of proofs $t_1 = \langle \pi_1^{(0)}, \ldots, \pi_1^{(k-1)} \rangle$ and $t_2 = \langle \pi_2^{(0)}, \ldots, \pi_2^{(k-1)} \rangle$. Then we compute $z_0 = \lambda$ and $z_{i+1} = V_A(z_i, \pi_1^{(i)}, \pi_2^{(i)})$ for each $0 \leq i < k$. If $z_i \underset{\neq}{\sqsubset} z_{i+1}$ for all i and $|z_k| > n$, then $V(x, t_1, t_2)$ outputs

$z_i[n]$. Otherwise, $V(x, t_1, t_2)$ outputs 0. The total length of the proofs is polynomial in n, so the total proof length is $2^{O(|x|)}$. Since V_A is polynomial-time, V runs in exponential-time and it follows that $L \in \mathsf{S}_2^{\mathsf{E}}$.

Let $L \in \mathsf{S}_2^{\mathsf{E}}$ and let V be a verifier for L as in Definition 2.4. On input z, we let $n = |z|$. We use V on input s_n to extend z by one bit. If $V(s_n, \pi_1, \pi_2) = 1$, we output $V'(z, \pi_1, \pi_2) = z1$. If $V(s_n, \pi_1, \pi_2) = 0$, we output $V'(z, \pi_1, \pi_2) = z0$. Then V' specifies a single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm because $|s_n|$ is logarithmic in $|z|$. We have $R(V') = L$, so $L \in R(\mathsf{FS}_2^{\mathsf{P}})$. □

We can now fully specify the symmetric alternation category notion used in this paper. We plug Lemma 3.1 into Definition 2.3:

1. X is $\mathsf{FS}_2^{\mathsf{P}}$-*meager* if player II has a winning strategy in the game $G[X; \mathsf{all}, \mathsf{FS}_2^{\mathsf{P}}]$.
2. X is $\mathsf{FS}_2^{\mathsf{P}}$-*comeager* if X^c is $\mathsf{FS}_2^{\mathsf{P}}$-meager.
3. X is *meager in* $\mathsf{S}_2^{\mathsf{E}}$ if $X \cap \mathsf{S}_2^{\mathsf{E}}$ is $\mathsf{FS}_2^{\mathsf{P}}$-meager.
4. X is *comeager in* $\mathsf{S}_2^{\mathsf{E}}$ if X^c is meager in $\mathsf{S}_2^{\mathsf{E}}$.

Lutz [14] proved that $\mathsf{SIZE}(\frac{2^n}{n})$ is meager in ESPACE where the strategy uses a brute force voting method inspired by Kannan [10]. We can replace this by Li's single-valued $\mathsf{FS}_2^{\mathsf{P}}$ algorithm from Theorem 2.7.

Theorem 3.2. $\mathsf{SIZE}(\frac{2^n}{n})$ *is* $\mathsf{FS}_2^{\mathsf{P}}$*-meager.*

Proof. Let $\mathcal{A}$ be the algorithm from Theorem 2.7. On input x, compute n so that $2^{n-1} \leq |x| \leq 2^n - 1$. Let $s(n) = \frac{2^n}{n}$. Our constructor δ outputs

$$\delta(x) = x0^{2^n - |x| - 1}\mathcal{A}(0^{2^n}).$$

The $0^{2^n-|x|-1}$ is to finish defining the language at length $n-1$. We then use $\mathcal{A}$ to define the language at length n. Thus $|\delta(x)| = 2^{n+1} - 1$ and $\delta(x)$ defines a subset of $\{0,1\}^{\leq n}$.

Let γ be any constructor. Then for infinitely many lengths n, $R(\gamma, \delta) = R(\delta \circ \gamma)$ does not have an $s(n)$-size circuit at length n. Therefore $R(\gamma, \delta) \notin \mathsf{SIZE}(s(n))$ and δ wins the Banach-Mazur game, so $\mathsf{SIZE}(s(n))$ is $\mathsf{FS}_2^{\mathsf{P}}$-meager. □

Theorem 1.3 is a corollary of Theorem 3.2, because $\mathsf{FS}_2^{\mathsf{P}}$-meager implies meager in $\mathsf{S}_2^{\mathsf{E}}$.

Theorem 1.3. $\mathsf{SIZE}(\frac{2^n}{n})$ *is meager in* $\mathsf{S}_2^{\mathsf{E}}$.

Our title result follows immediately from Theorem 1.3.

Corollary 3.3. *The class of problems requiring exponential-size circuits is comeager in* $\mathsf{S}_2^{\mathsf{E}}$.

Another corollary is meagerness of $\mathsf{SIZE}(\frac{2^n}{n})$ in E^{NP} under a derandomization hypothesis. This is because standard derandomization hypotheses [11] collapse $\mathsf{FS}_2^{\mathsf{P}}$ to $\mathsf{FP}^{\mathsf{NP}}$ via derandomization of $\mathsf{FZPP}^{\mathsf{NP}}$. Resource-bounded category in E^{NP} is defined analogously to category in E, using polynomial-time constructors that have access to an NP oracle in Definition 2.3.

Lemma 3.4. $R(\mathsf{FP}^{\mathsf{NP}}) = \mathsf{E}^{\mathsf{NP}}$.

1. X is $\mathsf{FP}^{\mathsf{NP}}$-*meager* if player II has a winning strategy in the game $G[X; \mathsf{all}, \mathsf{FP}^{\mathsf{NP}}]$.
2. X is *meager in* E^{NP} if $X \cap \mathsf{E}^{\mathsf{NP}}$ is $\mathsf{FP}^{\mathsf{NP}}$-meager.

Corollary 3.5. *If* E^{NP} *requires* $2^{\Omega(n)}$*-size* NP*-oracle circuits, then* $\mathsf{SIZE}(\frac{2^n}{n})$ *is meager in* E^{NP}.

Proof. Cai [4] showed that $\mathsf{S}_2^{\mathsf{P}} \subseteq \mathsf{ZPP}^{\mathsf{NP}}$. It also holds that single-valued $\mathsf{FS}_2^{\mathsf{P}}$ is contained in single-valued $\mathsf{FZPP}^{\mathsf{NP}}$ [6]. Under the hypothesis, techniques of Klivans and van Melkebeek [11] derandomize $\mathsf{FZPP}^{\mathsf{NP}}$ to $\mathsf{FP}^{\mathsf{NP}}$ [6]. Thus the constructor in the proof of Theorem 3.2 is in $\mathsf{FP}^{\mathsf{NP}}$, so $\mathsf{SIZE}(\frac{2^n}{n})$ is $\mathsf{FP}^{\mathsf{NP}}$-meager and is meager in E^{NP}. □

4 Conclusion

Lutz and Mayordomo [18] asked if P/poly has measure 0 in a class smaller than Δ_3^{E}, the third level of the exponential-time hierarchy. Theorem 1.3 implies P/poly is meager in $\mathsf{S}_2^{\mathsf{E}}$. Resource-bounded measure may also be defined in $\mathsf{S}_2^{\mathsf{E}}$ using single-valued $\mathsf{FS}_2^{\mathsf{P}}$ martingales. Does P/poly have measure 0 in $\mathsf{S}_2^{\mathsf{E}}$?

Acknowledgement. I thank an anonymous referee for helpful comments.

References

[1] S. Aaronson, B. Aydinlioglu, H. Buhrman, J. Hitchcock, and D. van Melkebeek. A note on exponential circuit lower bounds from derandomizing Arthur-Merlin games. Technical Report TR10-174, Electronic Colloquium on Computational Complexity, 2010. 1

[2] B. Aydinlioglu, D. Gutfreund, J. M. Hitchcock, and A. Kawachi. Derandomizing Arthur-Merlin games and approximate counting implies exponential-size lower bounds. *Computational Complexity*, 20(2):329–366, 2011. doi:10.1007/s00037-011-0010-8.

[3] Harry Buhrman, Lance Fortnow, and Thomas Thierauf. Nonrelativizing separations. In *Proceedings of the 13th Annual IEEE Conference on Computational Complexity*, pages 8–12. IEEE Computer Society, 1998. doi:10.1109/CCC.1998.694585. 1

[4] Jin-yi Cai. $\mathrm{S}_2^p \subseteq \mathsf{ZPP}^{\mathsf{NP}}$. *Journal of Computer and System Sciences*, 73(1):25–35, 2007. doi:10.1016/J.JCSS.2003.07.015. 2, 4

[5] Ran Canetti. More on BPP and the polynomial-time hierarchy. *Information Processing Letters*, 57(5):237–241, 1996. doi:10.1016/0020-0190(96)00016-6. 1, 2

[6] Lijie Chen, Shuichi Hirahara, and Hanlin Ren. Symmetric exponential time requires near-maximum circuit size. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC 2024, page 1990–1999, New York, NY, USA, 2024. Association for Computing Machinery. doi:10.1145/3618260.3649624. 1, 3, 4

[7] Gudmund Skovbjerg Frandsen and Peter Bro Miltersen. Reviewing bounds on the circuit size of the hardest functions. *Information Processing Letters*, 95(2):354–357, 2005. doi:10.1016/j.ipl.2005.03.009. 3

[8] J. M. Hitchcock, A. Sekoni, and H. Shafei. Counting martingales for measure and dimension in complexity classes. In *Proceedings of the 40th Computational Complexity Conference (CCC 2025)*, volume 339 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:35, 2025. doi:10.4230/LIPIcs.CCC.2025.20. 3

[9] J. M. Hitchcock and N. V. Vinodchandran. Dimension, entropy rates, and compression. *Journal of Computer and System Sciences*, 72(4):760–782, 2006. doi:10.1016/j.jcss.2005.10.002. 1

[10] R. Kannan. Circuit-size lower bounds and non-reducibility to sparse sets. *Information and Control*, 55(1-3):40–56, 1982. doi:10.1016/s0019-9958(82)90382-5. 1, 4

[11] A. Klivans and D. van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM Journal on Computing*, 31(5):1501–1526, 2002. doi:10.1137/s0097539700389652. 4

[12] Oliver Korten. The hardest explicit construction. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 433–444. IEEE, 2022. doi:10.1109/FOCS52979.2021.00051. 1, 3

[13] Zeyong Li. Symmetric exponential time requires near-maximum circuit size: Simplified, truly uniform. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC 2024, page 2000–2007, New York, NY, USA, 2024. Association for Computing Machinery. doi:10.1145/3618260.3649615. 1, 3

[14] J. H. Lutz. Resource-bounded Baire category and small circuits in exponential space. In *Proceedings of the Second Structure in Complexity Theory Conference*, pages 81–91. IEEE Computer Society Press, 1987. doi:10.1109/psct.1987.10319257. 1, 2, 4

[15] J. H. Lutz. *Resource-Bounded Category and Measure in Exponential Complexity Classes*. PhD thesis, California Institute of Technology, 1987. doi:10.7907/qny92-v6h14.

[16] J. H. Lutz. Category and measure in complexity classes. *SIAM Journal on Computing*, 19(6):1100–1131, 1990. doi:10.1137/0219076. 1, 2

[17] J. H. Lutz. Almost everywhere high nonuniform complexity. *Journal of Computer and System Sciences*, 44(2):220–258, 1992. doi:10.1016/0022-0000(92)90020-j. 2, 3

[18] J. H. Lutz and E. Mayordomo. Twelve problems in resource-bounded measure. In G. Păun, G. Rozenberg, and A. Salomaa, editors, *Current Trends in Theoretical Computer Science: Entering the 21st Century*, pages 83–101. World Scientific Publishing, 2001. doi:10.1142/9789812810403_0001. 5

[19] E. Mayordomo. Almost every set in exponential time is P-bi-immune. *Theoretical Computer Science*, 136(2):487–506, 1994. doi:10.1016/0304-3975(94)00023-c. 1

[20] P. B. Miltersen, N. V. Vinodchandran, and O. Watanabe. Superpolynomial versus subexponential circuit size in the exponential hierarchy. In *Proceedings of the Fifth Annual International Computing and Combinatorics Conference*, pages 210–220, 1999. doi:10.1007/3-540-48686-0_21. 1

[21] Alexander Russell and Ravi Sundaram. Symmetric alternation captures BPP. *Comput. Complex.*, 7(2):152–162, 1998. doi:10.1007/S000370050007. 1, 2

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il