Hans Journal of Computational Biology 计算生物学, 2025, 15(4), 37-46
Published Online December 2025 in Hans. https://www.hanspub.org/journal/hjcb
https://doi.org/10.12677/hjcb.2025.154004

Hans汉斯

细菌中排序算法实现研究

陈　梅*

中央民族大学信息工程学院，北京

收稿日期：2025年10月19日；录用日期：2025年11月12日；发布日期：2025年11月19日

摘　要

排序算法在信息科学中占据着不可替代的地位。细胞信息处理设备在生物治疗、生物修复等多个领域执行计算任务，然而活体内的排序算法研究仍属空白。本研究通过将经典算法设计策略与先进基因技术相结合，成功在大肠杆菌中设计并验证了一种遗传排序算法。该算法基于分治策略设计，利用CRISPR/Cas9系统通过递归方式将信号划分为较大组与较小组实现排序，结果经由第二代测序技术进行分析。湿实验验证了该排序算法的有效性。此外，算法中提供的分组方法亦代表了一种高效的分类机制。本研究通过排序与分类的基因电路实现来推动智能复杂细胞信息处理设备的构建。

关键词

生物计算，细胞计算，DNA计算，排序，合成生物学

Developing a Sorting Algorithm in Bacteria

Mei Chen*

School of Information Engineering, Minzu University of China, Beijing

Received: October 19, 2025; accepted: November 12, 2025; published: November 19, 2025

Abstract

The sorting algorithm occupies an irreplaceable position in information science. Cellular information-processing devices execute computing tasks in various fields, such as biotherapy and bioremediation. However, the sorting algorithm in vivo is still lacking. Through combining classical algorithm-design strategy with advanced genetic technologies, this study has successfully designed and tested a genetic sorting algorithm in the bacterium Escherichia coli. This sorting algorithm is designed based on the divide-and-conquer algorithm-design strategy. It enables sorting through a recursive grouping of the signals into the greater group and the lesser group with clustered regularly interspaced short palindromic repeats/CRISPR-associated9 system. The result is analyzed by next-generation sequencing. Wet-lab experiments verified the validity of this sorting algorithm. In

*通讯作者。

文章引用: 陈梅. 细菌中排序算法实现研究[J]. 计算生物学, 2025, 15(4): 37-46. DOI: 10.12677/hjcb.2025.154004

addition, the grouping method provided in this sorting algorithm also represents an efficient classification method. This study will advance the construction of intelligent and complex cellular information-processing devices through sorting and classification.

Keywords

Bio-Computing, Cellular Computing, DNA Computing, Sorting, Synthetic Biology

1. 引言

生物技术的发展使研究人员能够合理操控遗传信息流，构建细胞信息处理设备。这类设备因其可与自然界交互执行计算任务(如生物治疗、生物修复等领域)，具有广泛的应用前景。例如，当检测到两种特定肿瘤标志物时，用于肿瘤治疗的细胞“与门”会生成组合免疫调节输出以激活治疗响应[1]。受电子计算机强大信息处理能力的启发，科研人员已构建出众多细胞信息处理设备。这些设备主要聚焦于模拟电子计算功能，包括触发开关[2]、振荡器[3]、逻辑门[4]-[11]、传感器[12]、计数器[13]、交换器[14]、存储器[15]-[21]及逻辑单元[22]-[24]。这些设备将计算科学的信息处理思想与体内信息处理方法相结合，展现出细胞计算设备的巨大潜力。

排序算法作为信息处理中的核心操作之一，在数据组织、模式识别与决策优化中发挥着不可替代的作用。它能够将杂乱无章的数据按升序或降序排列，从而为后续分析提供结构化基础。在细胞信息处理设备中，因为细胞内资源有限，排序算法能够甄别信号优先级，引导细胞将有限的资源投入到最关键的生理响应中，从而实现决策智能化与资源利用最优化。然而在活体内开发排序算法面临诸多挑战：需要对多个信号进行大量比较运算并记录结果。由于细胞体积微小且资源有限，必须通过精巧设计和高效工具来降低实现复杂度。

本研究开发了一种细菌排序算法(SAB)。该算法基于分治策略设计，通过 CRISPR/Cas9 系统与第二代测序技术实现，并经由湿实验验证了其有效性。

2. 结果

2.1. 整体设计

为实现信号排序，本研究为每个信号设定特定短 DNA 序列作为特征序列。细菌排序算法通过操控这些特征序列来实现信号处理。该算法基于分治策略进行设计。分治策略是最经典的算法设计策略之一，其核心思想是将原问题递归分解为两个或多个相互独立的子问题，直至子问题简化为可直接求解的形式，最后将子问题的解合并形成原问题的解。本算法首先将所有待处理信号划分为较大组和较小组；随后对每个分组继续执行相同的二分操作，直至组内仅剩单个信号。最终通过整合所有分组结果即可呈现信号的排序(图 1 和图 S1)。

2.2. 分组系统设计

CRISPR/Cas9 系统在本细菌排序算法中作为分组工具使用。作为一项革命性的基因组编辑技术，

CRISPR/Cas9 系统已获得广泛应用。该系统通过 20 bp 的 sgRNA 引导 Cas9 核酸酶靶向具有 PAM 序列(5'-NGG-3')的 DNA 位点，由 Cas9 在靶标位点催化产生 DNA 双链断裂[25]。质粒发生双链断裂通常会导致其自身丢失，因此通过诱导质粒产生双链断裂成为清除质粒的有效手段。本算法利用 CRISPR/Cas9 系统通过诱导双链断裂清除质粒来实现分组功能。研究设计了三质粒系统以递归方式完成信号分组(图 2)。该分组过程通过清除含有较小组信号特征序列的质粒来实现分组，最终通过特征序列是否被清除来判定信号所属组别。

本研究采用第二代测序技术对细菌排序算法的结果进行分析。该技术是一种大规模并行测序方法，可同时对数百万乃至数十亿条 DNA 链进行测序[26]。考虑到 pL 质粒可能无法被完全清除，现定义 K 值用于结果分析如下：

$$K\left(\frac{A}{B}\right)=\frac{\text{信号}A\text{特征序列数}}{\text{信号}B\text{特征序列数}}$$

通过二代测序技术对特征序列进行定量分析，并依据 K 值的变化情况判定最终排序结果(图 3 和图 S2)。

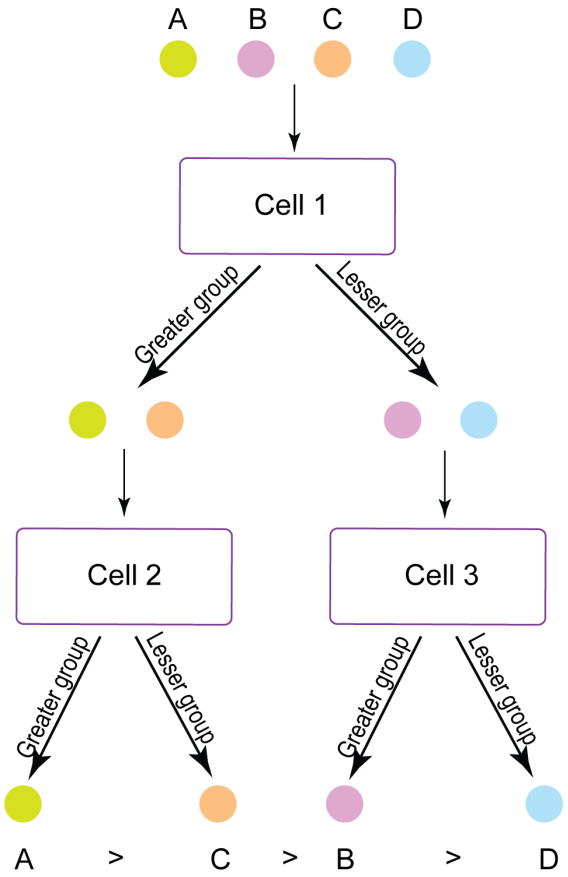

Figure 1. Schematic representation of the sorting principle. Multiple independent types of cells need to be constructed to enable sorting. In these cells, some group all signals, while some group signals in a certain group are divided by another cell. The cells are used in a pre-determined order to perform sorting

图 1. 排序原理示意图。为实现排序功能，需要构建多种独立类型的细胞。这些细胞中，部分负责对所有信号进行分组，另一部分则对经其他细胞划分的特定信号组进行再分组。所有细胞按预定顺序执行排序操作

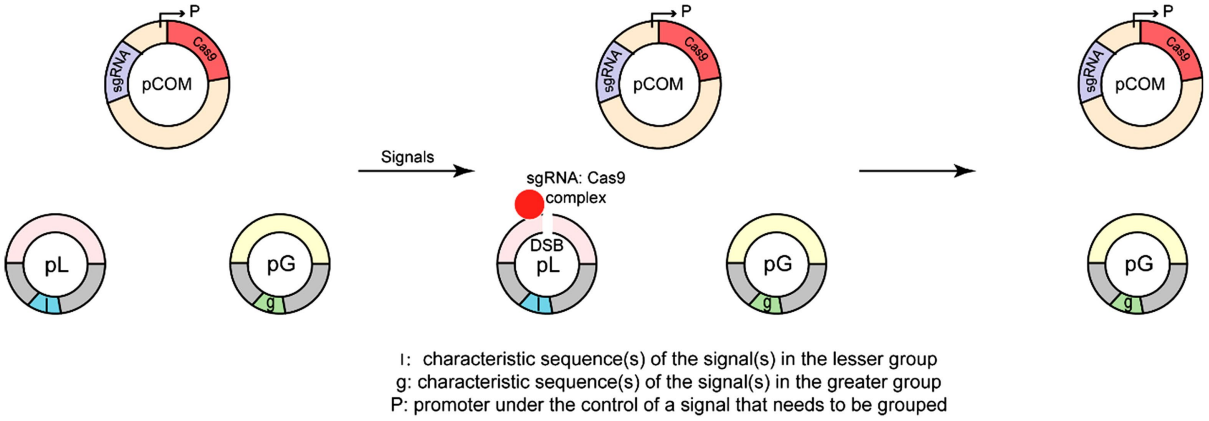

Figure 2. Schematic representation of the three-plasmid system. The three-plasmid system is composed of plasmid pG, pL, and pCOM. Plasmid pG contains the characteristic sequences of the signals in the greater group. Plasmid pL contains the characteristic sequences of the signals in the lesser group. Plasmid pCOM produces sgRNA: Cas9 complex, for causing a double-stranded break (DSB) in the pL under the control of a signal that needs to be grouped

图 2. 三质粒系统示意图。该体系由 pG、pL 和 pCOM 三种质粒构成：pG 质粒携带较大组信号的特征序列；pL 质粒携带较小组信号的特征序列；pCOM 质粒在待分组信号调控下表达 sgRNA:Cas9 复合体，用于在 pL 质粒上引发 DNA 双链断裂

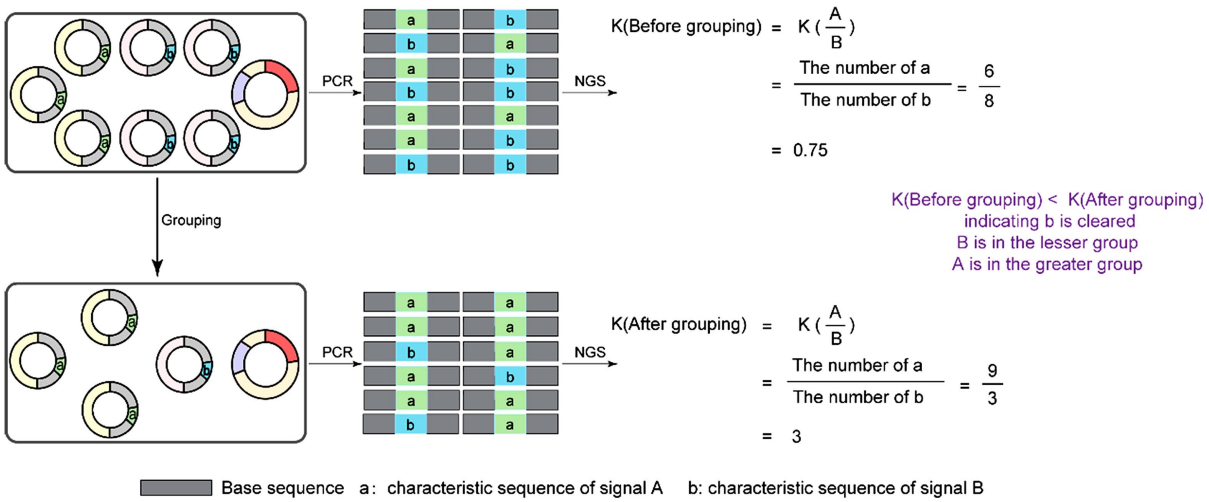

Figure 3. Schematic representation of the result analysis. To analyse the result, a fragment of base sequence is designed and the characteristic sequences are inserted into the base sequence. Before and after grouping, polymerase chain reaction (PCR) is performed to obtain the characteristic sequences. The PCR products are sequenced and counted using NGS. K value is computed according to the counting result. The change in K value indicates the grouping result. In the grouping process, the number of plasmids containing the characteristic sequence of the lesser signal dramatically decreases, even if not totally cleared. Therefore, the K value changes dramatically. The clearance of the characteristic sequence is determined according to this change

图 3. 结果分析示意图。为进行结果分析，首先设计基础碱基序列片段并将特征序列插入其中。在分组操作前后，通过聚合酶链式反应(PCR)获取特征序列，对 PCR 产物进行二代测序(NGS)和定量分析，根据计数结果计算 K 值。K 值的变化用于判定分组结果：在分组过程中，即使未被完全清除，携带较弱信号特征序列的质粒数量也会急剧减少，这种显著的数量变化导致 K 值发生剧烈波动，最终根据该波动判定特征序列的清除情况

2.3. 湿实验

本研究利用细菌排序算法，在大肠杆菌中对阿拉伯糖(Arab)、异丙基-β-D-硫代半乳糖苷(IPTG)、无水四环素(aTc)及鼠李糖(Rham)四种信号分子进行了排序。假设四种信号的强度关系为 Arab > Rham > IPTG > aTc，图 4 展示了对应的基因电路设计及排序结果。

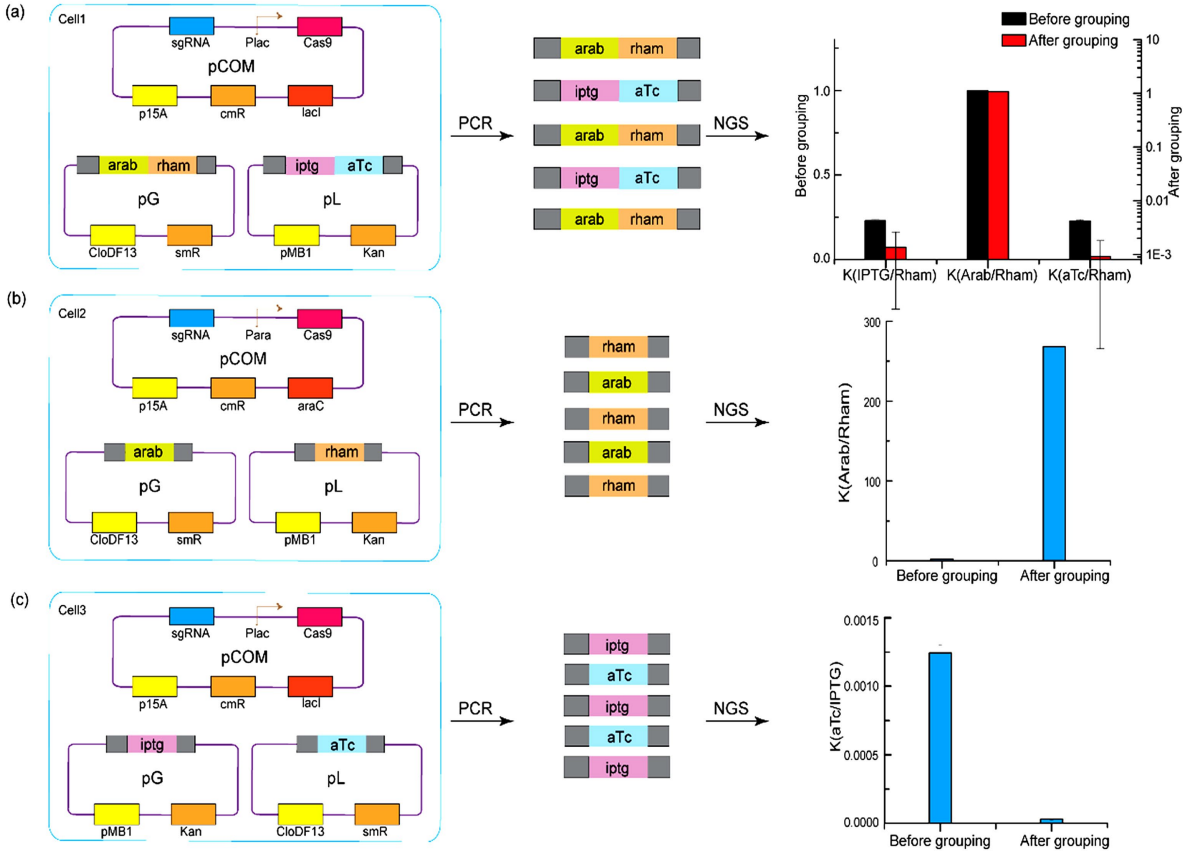

Figure 4. Sorting four signals using bacteria. (a) Cell1 divided the four signals into the greater group and the lesser group. The K value of arabinose and rhamnose remained the same before and after the grouping process, indicating that the two signals are in one group and the other two signals are in the other group. The K value of IPTG and rhamnose significantly decreased, indicating that the IPTG is cleared. The K value of aTc and rhamnose also significantly decreased, indicating that the aTc is cleared. Consequently, the results demonstrated that the greater group contained arabinose and rhamnose, and the lesser group contained IPTG and aTc. (b) Cell2 grouped the signals in the greater group. The K value of arabinose and rhamnose significantly increased, indicating that rhamnose is cleared. Consequently, arabinose is greater than rhamnose. (c) Cell3 grouped the signals in the lesser group. The K value of aTc and IPTG significantly decreased, indicating that the aTc is cleared. Consequently, IPTG is greater than aTc. Combining the results of cell1 and cell2 with cell3, the order is Arab > Rham > IPTG > aTc. Values and error bars represent the mean ± SD of three replicates

图 4. 利用细菌对四种信号进行排序。(a) Cell1 将四种信号划分为较大组和较小组。阿拉伯糖和鼠李糖的 K 值在分组前后保持稳定，表明这两种信号属于同一分组，而另外两种信号属于另一分组。IPTG 和鼠李糖的 K 值显著下降，表明 IPTG 被清除；aTc 和鼠李糖的 K 值也显著下降，表明 aTc 被清除。由此证明较大组包含阿拉伯糖和鼠李糖，较小组包含 IPTG 和 aTc。(b) Cell2 对较大组信号进行再分组。阿拉伯糖与鼠李糖的 K 值显著上升，表明鼠李糖被清除，因此阿拉伯糖强度大于鼠李糖。(c) Cell3 对较小组信号进行再分组。aTc 与 IPTG 的 K 值显著下降，表明 aTc 被清除，因此 IPTG 强度大于 aTc。综合 cell1、cell2 与 cell3 的实验结果，最终排序为：阿拉伯糖 > 鼠李糖 > IPTG > aTc。图中数据及误差线表示三次重复实验的平均值 ± 标准差

为实现排序功能，本研究设计了三种工程细胞。每种细胞均包含一个三质粒系统。为确保三质粒系统的稳定存在并便于菌株构建，每种质粒均采用不同的复制起点(图 4 中 p15A、CloDF13 和 pMB1)和不同抗性标记(图 4 中 cmR、smR 和 Kan)。同类质粒在不同种类细胞中(如 cell1 与 cell3 中的 pG 质粒)可具有不同的复制起点和抗性标记。在 cell1 和 cell3 中，CRISPR/Cas9 系统的靶序列为 5′-gacctgccctgtgcagctgtggg-3′，而 cell2 中则为 5′-aactgcaatttattcatatcagg-3′。Cell1 和 cell3 采用 IPTG 诱导型表达系统执行分组操作，cell2 则使用阿拉伯糖诱导型表达系统。表 1 列出了特征序列及其碱基序列，表 2 展示了测序计数结果。

Table 1. The characteristic sequences and base sequences (5′-3′)
表 1. 特征序列和基础序列(5′-3′)

特征序列	arabinose	aactgcaatttattcacgat
	rhamnose	aactgcaatttattcatatc
	IPTG	gacctgccctgtgcagcacc
	aTc	gacctgccctgtgcagctgt
基础序列(PCR 和 NGS 中使用的引物用斜体指示，“x”指示插入的特征序列)	Cell1	aagtttatctgtacaacaggcaagctgcccgttccctggcc*taccttggtcaccacactcac-ttatggagtgcaatgcttcgctcgttatcccgaccxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxacatgaagcagcacgatttttcaagtccgcgatgcccgagggctacgttcaa-gaacgcactatagttttaaggacgacgggacatataagacccgtgccgaggtgaagtttgaag**gcgatactttggtgaacagaat*tgagctgaaggggatcgacttttaaagaagacggaaatatcctgggacacaagctg-gagtataacttcaatagccataacgtttacattacagctgacaaacagaaaaatggtataaaagccaacttcaagatccggcataatgttgaagatggcagcgtgcagctggctgat
	Cell2	ttcggcctattggttaaaaaatgagctgatttaacaaaaatttaacgcgaattttaacaaaa-tattaacgtttacaatttcaggtggcacttttcggggaaatgtgcgcggaaccctatttgtttatttttctaa*atacattcaaaatatgtatccgctca*tgaattaattcttagaaaaactcatcgag-catcaaatgaxxxxxxxxxxxxxxxxxxxxxxxxxxattatcaataccatatttttgaaaaagccgtttctgtaatgaaggagaaaactcaccgaggcagttcc*ataggatggcaagatcctggtatc*ggtctgcgat-tccgactcgtccaacatcaatacaacctattaatttcccctcgtcaaaaataaggttatcaagtgagaaatcaccatgagtgacgactgaatccggtg
	Cell3	aagtttatctgtacaacaggcaagctgcccgttccctggcc*taccttggtcaccacactcac-ttatggagtgcaatgcttcgctcgttatcccgaccxxxxxxxxxxxxxxxxxxxxxxxxxxacatgaagcagcacgatttttcaagtccgcgatgcccgagggctacgttcaagaacgcactataagttttaaggacgacgg-gacatataagacccgtgccgaggtgaagtttgaag**gcgatactttggtgaacagaat*tgagctgaaggggatcgacttttaaagaagacggaaatatcctgggacacaagctggagtataacttcaatagccataac-gtttacattacagctgacaaacagaaaaatggtataaaagccaacttcaagatccggcataatgttgaagatggcagcgtgcagctggctgat

Table 2. The counting results
表 2. 计数结果

	Cell	Characteristic sequence	Replicate 1	Replicate 2	Replicate 3
Before grouping	Cell1	Rham	46022	45801	46641
		Arab	45870	45601	46449
		aTc	10661	10658	10212
		IPTG	10735	10708	10256
	Cell2	Rham	770707	219871	420490
		Arab	1998927	562992	1025309
	Cell3	aTc	4124	3641	3668
		IPTG	3322351	3061058	2811369
After grouping	Cell1	Rham	3176	2995	2806
		Arab	3427	3223	2991
		aTc	3	0	5
		IPTG	5	0	7
	Cell2	Rham	5818	4606	4890
		Arab	1634990	1165340	1318921
	Cell3	aTc	85	86	119
		IPTG	3290052	2718175	3435341

3. 结论和讨论

本研究开发了一种可扩展的细菌排序算法。首先，该算法架构具有可扩展性：细菌排序算法中每种细胞类型均可独立执行分组操作，便于灵活添加细胞单元。与传统的体内分布式计算系统相比，基于分治策略的算法架构避免了使用化学导线(依赖特定分子在细胞间传递信息)，有效降低了设计难度。其次，分组细胞单元具备可扩展性：通过构建三质粒系统在每个细胞内递归执行分组，降低了实现复杂度。该基于 CRISPR/Cas9 系统的三质粒设计不受信号数量影响，质粒数量不随信号规模增加而改变，仅需扩展特征序列即可处理更多信号。根据质粒容量估算，该算法具备处理数十种信号的潜力。

该细菌排序算法通过提供灵活分类方法，将推动智能细胞信息处理设备的构建。分类是众多智能任务的基础，本算法提供的分组机制代表了一种高效分类方法。三质粒系统中 pG 与 pL 质粒承载的信号数量无需对等，支持对称与非对称分组模式(图 S3)，使细胞能按类别处理信号：可优先处理重要信号(较大组)，或忽略次要信号(较小组)。同时，细胞可参照同组其他信号特征进行决策，基于群体特征预测作出判断。

通过引入算法设计策略，该研究还将促进复杂细胞信息处理设备的开发。细胞的信息处理模式与电子设备存在本质差异[27]：例如细胞不像电子设备那样通过级联布尔逻辑门执行复杂功能。简单模仿电子设备的工作方式并非编程细胞的最佳途径，因此探索新的构建策略至关重要。算法在发挥电子计算机信息处理能力应用于各领域时具有不可替代的作用——电子计算机硬件仅执行特定逻辑运算，而算法根据任务特性和计算硬件特点，提供完成任务的方法步骤。正是算法指导计算机实现人脸识别、商品推荐、三维渲染等多样化任务。我们期待本研究能推动基于细胞本质特性的算法开发，为实现更复杂功能奠定基础。

尽管该细菌排序算法具备很多优势，但其实际应用仍面临细胞内代谢负荷、遗传稳定性及计算效率等现实挑战。递归性三质粒系统和 CRISPR/Cas9 元件的持续表达可能引发生长迟滞，限制其处理效率；依赖荧光报告等生物读出方法，存在速度慢、通量低和成本高的问题。为突破这些瓶颈，未来改进可聚焦于：通过基因组整合或研发新型 Cas 蛋白等工具来增强系统稳定性；结合微流控或新型测序技术开发高通量、快速的电/光信号读出方法，从而向更高效、智能的生物信息处理系统演进。

4. 方法

完成设计后，质粒由重庆英茂盛业生物科技有限公司合成。选择大肠杆菌 BL21(DE3)菌株(F^- ompT hsdSB (rB-mB-) gal dcm (DE3))作为 cell1 和 cell3 的宿主，大肠杆菌 MG1655 菌株(F^- λ^- ilvG$^-$ rfb-50 rph-1)作为 cell2 的宿主。对于每种细胞，首先共转化 pG 与 pL 质粒，随后转化 pCOM 质粒，采用标准热激转化法。所有细胞均在 LB 培养基中培养。

排序实验流程如下：初始阶段将细胞接种于含 25 μg/mL 氯霉素、50 μg/mL 卡那霉素和 50 μg/mL 链霉素的 LB 培养基，37℃振荡培养过夜后，按 5 μL 菌液转入 5 mL 新鲜培养基的比例进行稀释，新培养基在保留氯霉素的同时添加特定信号分子。细胞继续在 37℃培养 12-16 小时。各信号分子终浓度为：无水四环素 100 ng/mL、阿拉伯糖 0.2%、IPTG 0.1 mM、鼠李糖 0.1 mg/mL。

针对每种细胞，通过 PCR 扩增碱基序列，扩增产物采用二代测序技术进行测序。根据测序结果统计各信号特征序列的数量。Cell1 和 cell2 的测序及计数工作由北京君唯诺生物技术有限公司完成，cell3 由北京阅微基因科技有限公司完成。

参考文献

[1] Nissim, L., Wu, M., Pery, E., Binder-Nissim, A., Suzuki, H.I., Stupp, D., *et al*. (2017) Synthetic RNA-Based Immunomodulatory Gene Circuits for Cancer Immunotherapy. *Cell*, **171**, 1138-1150.e15. https://doi.org/10.1016/j.cell.2017.09.049

[2] Gardner, T.S., Cantor, C.R. and Collins, J.J. (2000) Construction of a Genetic Toggle Switch in *Escherichia coli*. *Nature*, **403**, 339-342. https://doi.org/10.1038/35002131

[3] Elowitz, M.B. and Leibler, S. (2000) A Synthetic Oscillatory Network of Transcriptional Regulators. *Nature*, **403**, 335-338. https://doi.org/10.1038/35002125

[4] Moon, T.S., Lou, C.B., Tamsir, A., Stanton, B.C. and Voigt, C.A. (2012) Genetic Programs Constructed from Layered Logic Gates in Single Cells. *Nature*, **491**, 249-253. https://doi.org/10.1038/nature11516

[5] Bonnet, J., Yin, P., Ortiz, M.E., Subsoontorn, P. and Endy, D. (2013) Amplifying Genetic Logic Gates. *Science*, **340**, 599-603. https://doi.org/10.1126/science.1232758

[6] Kim, H., Bojar, D. and Fussenegger, M. (2019) A Crispr/Cas9-Based Central Processing Unit to Program Complex Logic Computation in Human Cells. *Proceedings of the National Academy of Sciences*, **116**, 7214-7219. https://doi.org/10.1073/pnas.1821740116

[7] Xie, Z., Wroblewska, L., Prochazka, L., Weiss, R. and Benenson, Y. (2011) Multi-Input RNAi-Based Logic Circuit for Identification of Specific Cancer Cells. *Science*, **333**, 1307-1311. https://doi.org/10.1126/science.1205527

[8] Green, A.A., Kim, J., Ma, D., Silver, P.A., Collins, J.J. and Yin, P. (2017) Complex Cellular Logic Computation Using Ribocomputing Devices. *Nature*, **548**, 117-121. https://doi.org/10.1038/nature23271

[9] Tamsir, A., Tabor, J.J. and Voigt, C.A. (2011) Robust Multicellular Computing Using Genetically Encoded NOR Gates and Chemical 'Wires'. *Nature*, **469**, 212-215. https://doi.org/10.1038/nature09565

[10] Regot, S., Macia, J., Conde, N., Furukawa, K., Kjellén, J., Peeters, T., *et al*. (2011) Distributed Biological Computation with Multicellular Engineered Networks. *Nature*, **469**, 207-211. https://doi.org/10.1038/nature09679

[11] Gander, M.W., Vrana, J.D., Voje, W.E., Carothers, J.M. and Klavins, E. (2017) Digital Logic Circuits in Yeast with Crispr-Dcas9 NOR Gates. *Nature Communications*, **8**, Article No. 15459. https://doi.org/10.1038/ncomms15459

[12] Levskaya, A., Chevalier, A.A., Tabor, J.J., *et al*. (2005) Engineering *Escherichia coli* to See Light. *Nature*, **438**, 441-442. https://doi.org/10.1038/nature04405

[13] Friedland, A.E., Lu, T.K., Wang, X., Shi, D., Church, G. and Collins, J.J. (2009) Synthetic Gene Networks That Count. *Science*, **324**, 1199-1202. https://doi.org/10.1126/science.1172005

[14] Callura, J.M., Cantor, C.R. and Collins, J.J. (2012) Genetic Switchboard for Synthetic Biology Applications. *Proceedings of the National Academy of Sciences*, **109**, 5850-5855. https://doi.org/10.1073/pnas.1203808109

[15] Shipman, S.L., Nivala, J., Macklis, J.D. and Church, G.M. (2016) Molecular Recordings by Directed CRISPR Spacer Acquisition. *Science*, **353**, aaf1175. https://doi.org/10.1126/science.aaf1175

[16] Perli, S.D., Cui, C.H. and Lu, T.K. (2016) Continuous Genetic Recording with Self-Targeting Crispr-Cas in Human Cells. *Science*, **353**, aag0511. https://doi.org/10.1126/science.aag0511

[17] Farzadfard, F. and Lu, T.K. (2014) Genomically Encoded Analog Memory with Precise *in Vivo* DNA Writing in Living Cell Populations. *Science*, **346**, Article 1256272. https://doi.org/10.1126/science.1256272

[18] Sheth, R.U., Yim, S.S., Wu, F.L. and Wang, H.H. (2017) Multiplex Recording of Cellular Events over Time on CRISPR Biological Tape. *Science*, **358**, 1457-1461. https://doi.org/10.1126/science.aao0958

[19] Shipman, S.L., Nivala, J., Macklis, J.D. and Church, G.M. (2017) Crispr-Cas Encoding of a Digital Movie into the Genomes of a Population of Living Bacteria. *Nature*, **547**, 345-349. https://doi.org/10.1038/nature23017

[20] Tang, W. and Liu, D.R. (2018) Rewritable Multi-Event Analog Recording in Bacterial and Mammalian Cells. *Science*, **360**, eaap8992. https://doi.org/10.1126/science.aap8992

[21] Chen, W.G., Han, M.Z., Zhou, J.T., *et al*. (2021) An Artificial Chromosome for Data Storage. *National Science Review*, **8**, nwab028. https://doi.org/10.1093/nsr/nwab028

[22] Ausländer, S., Ausländer, D., Müller, M., Wieland, M. and Fussenegger, M. (2012) Programmable Single-Cell Mammalian Biocomputers. *Nature*, **487**, 123-127. https://doi.org/10.1038/nature11149

[23] Choi, S., Lee, G. and Kim, J. (2022) Cellular Computational Logic Using Toehold Switches. *International Journal of Molecular Sciences*, **23**, Article 4265. https://doi.org/10.3390/ijms23084265

[24] Shao, J., Qiu, X., Zhang, L., Li, S., Xue, S., Si, Y., *et al*. (2024) Multi-Layered Computational Gene Networks by Engineered Tristate Logics. *Cell*, **187**, 5064-5080.e14. https://doi.org/10.1016/j.cell.2024.07.001

[25] Jinek, M., Chylinski, K., Fonfara, I., Hauer, M., Doudna, J.A. and Charpentier, E. (2012) A Programmable Dual-RNA-Guided DNA Endonuclease in Adaptive Bacterial Immunity. *Science*, **337**, 816-821. https://doi.org/10.1126/science.1225829

[26] Goodwin, S., McPherson, J.D. and McCombie, W.R. (2016) Coming of Age: Ten Years of Next-Generation Sequencing Technologies. *Nature Reviews Genetics*, **17**, 333-351. https://doi.org/10.1038/nrg.2016.49

[27] Xu, J. (2016) Probe Machine. *IEEE Transactions on Neural Networks and Learning Systems*, **27**, 1405-1416. https://doi.org/10.1109/tnnls.2016.2555845

附录：补充材料

```
Grouping(Signals[ ])
{
   k=lenth(Signals[ ])

   if k==1
      return

   else
   {
      m=⌈k/2⌉

      put the biggest m signals into
the greater_group[ ]
      put the rest k-m signals into the
lesser_group[ ]

      Grouping(greater_group[ ])
      Grouping(lesser_group[ ])
    }

}
```

Figure S1. The pseudo-code description of SAB
图 S1. SAB 伪代码

```
Signals[n]

number_before[n]  //The number of characteristc
                  // sequences of all signals before grouping
number_after[n]   //The number of characteristc
                  // sequences of all signals after grouping

//Group1,group2 are used to store the grouping results
group1[ ]
group2[ ]
group1[0]=Signals[n-1]

Flag=0 //Flag indicates the relationship of group1 and group2

K_before[n-1]
K_after[n-1]

//Computing the K value
for (i=0,i<n-1,i++)
      K_before[i]= number_before[i] / number_before[n-1]
for (i=0,i<n-1,i++)
      K_after[i]= number_after[i] / number_after[n-1]

//Grouping
for (i=0,i<n-1,i++)
      if (K_before[i]<K_after[i])
          Add Signals[i] to group2[ ]
          Flag=1
      if (K_before[i]>K_after[i])
          Add Signals[i] to group2[ ]
      if (K_before[i]==K_after[i])
          Add Signals[i] to group1[ ]

//Determining the relationship
if (Flag==0)
      copy group1[ ] to the greater_group[ ]
      copy group2[ ] to the lesser_group[ ]
if (Flag==1)
      copy group1[ ] to the lesser_group[ ]
      copy group2[ ] to the greater_group[ ]
```

Figure S2. The pseudo-code description of grouping process
图 S2. 分组过程伪代码

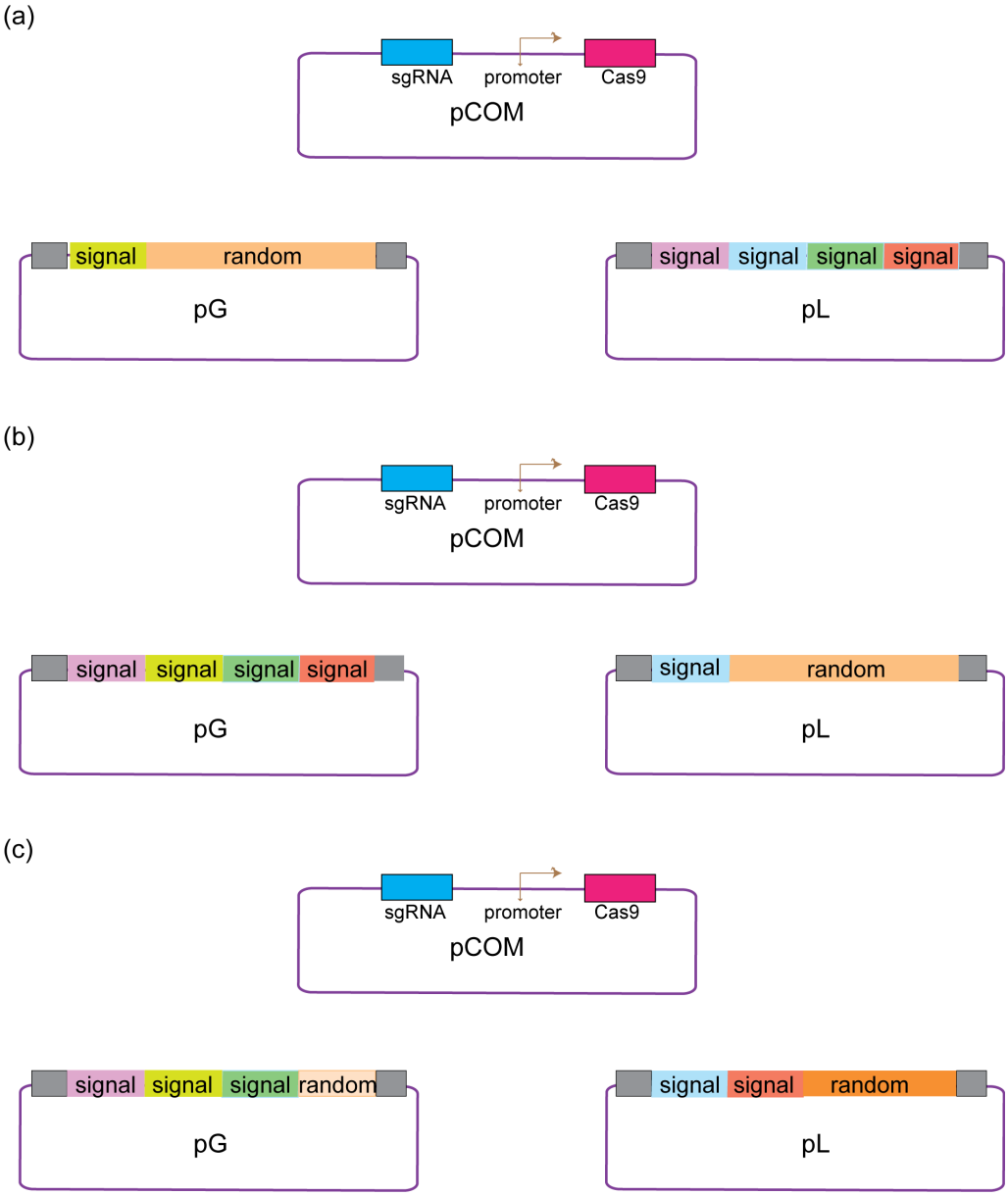

Figure S3. Schematic representation of classification in asymmetric ways (random sequence is used to ensure that the base sequences in pL and pG are of the equal length). (a) Finding the maximum. (b) Finding the minimum. (c) Dividing five signals into two groups

图 S3. 非对称分类示意图（使用随机序列确保 pL 和 pG 中的碱基序列长度相等）。(a) 寻找最大值。(b) 寻找最小值。(c) 将五个信号分成两组