Reflective Report: LAMP Primer Auto Design Tool

Group mini-project 9 - Guobiao YE 2074

***Abstract*—Loop-mediated Isothermal Amplification (LAMP) is a molecular diagnostic technique requiring complex primer design. This report introduces the LAMP Primer Auto Design (LPAD) tool, an automated system for designing, filtering, and scoring LAMP primers. LPAD processes target and background sequences to produce ranked primer sets optimized for specificity and efficiency. Divided into predesign, filtration, and scoring modules, LPAD incorporates mutation filtering and penalty-based scoring to enhance primer quality. Benchmarking shows LPAD delivers results comparable to PrimerExplorer5, with added features like loop primer inclusion and improved specificity. While some scalability and parameter optimization challenges remain, LPAD demonstrates strong potential as a user-friendly tool for researchers and clinicians.**

I. Introduction

LOOP-MEDIATED Isothermal Amplification (LAMP) is a nucleic acid amplification technique based on isothermal conditions, widely used in molecular diagnostics and genetic testing (Wong et al. 2018). Unlike traditional Polymerase Chain Reaction (PCR), LAMP performs amplification under constant temperature, offering advantages such as simplicity, speed, and high sensitivity, making it especially suitable for point-of-care testing (Soroka et al. 2021).

The design of LAMP primers is a critical step in the process. Unlike PCR, which uses a pair of primers, LAMP requires at least four different primers: Forward Outer Primer, Backward Outer Primer, Forward Inner Primer (FIP), and Backward Inner Primer (BIP) (Fig 1). These primers form specific loop structures through complementary pairing, initiating the amplification reaction (Notomi et al. 2015). Additionally, further primers (Loop F and Loop B) could be designed to further accelerate the amplification efficiency. The primer design must be tailored to the target gene sequence by a series of constraints, ensuring specificity and efficiency (Notomi et al. 2000). Multiple primers come with complex calculations, posing a challenge for primer design and selection.

Existing LAMP primer design tools lack a fully automated process from input to output, and the results are not unique. It requires users to manually set the parameters and select primers based on experience, increasing the threshold for tool use. Additionally, the specificity of the primers against the background genomes/transcriptomes is not considered, which may lead to amplification of non-target sequences and result in contamination. Therefore, we developed the LAMP Primer Auto Design (LPAD) Tool, which automates primer design, specificity filtration and scoring. By passing the target sequence and background genome/transcriptome to the LPAD, the user can directly get the possible LAMP primers sorted by score. To broaden the usage scenarios, we also allow users to manually change the built-in parameters for custom designs. The LPAD streamlines the primer design process, reduces user dependency on manual adjustments, and enhances the accuracy of primer selection, making it more accessible to both experienced and novice users.

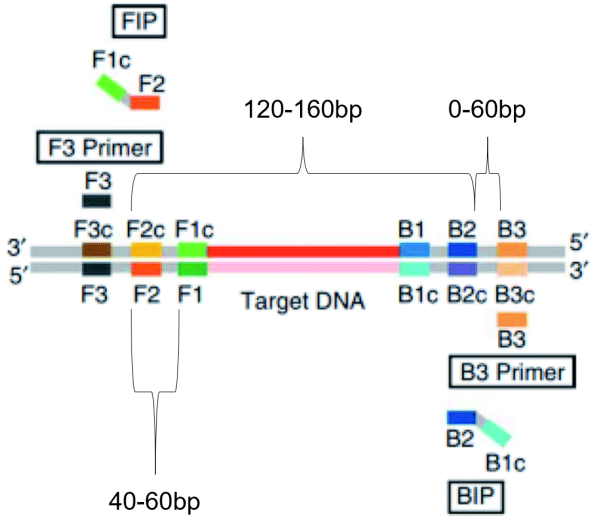

Fig. 1. **Primer design for the LAMP reaction.** The FIP consists of the outer region, which binds to the complementary F2 on the template, and the inner region, which binds to the complementary F1c on the newly synthesized strand. Similarly, the outer region of BIP targeting B2 on the template, and the inner region, targeting B1c on the newly formed strand. The Forward External Primer (F3) and Backward External Primer (B3) specifically bind to the F3 and B3 regions of the template, respectively. Additionally, the Forward Loop Primer (FL) and Backward Loop Primer (BL) bind to the single-stranded loop regions, enhancing amplification efficiency. Adapted from Notomi et al. (2015).

II. Methods and Results

Our goal is to develop an automated system that generates ranked, specific LAMP primer sets based on a given sequence and background genome/transcriptome. This system enables users to efficiently select the most suitable primers by directly referring to their scores. To achieve this, we divided the program into three modules:

(1) Predesigning potential LAMP primers;

(2) Aligning candidate primers against the background file to filter out non-specific ones;

(3) Scoring and ranking the filtered LAMP primers.

A. Predesign of Primers

Primers predesigning is the core of the procedure, where we developed a multi-step algorithm that integrates primer design, variability filtering, and combination processes. The method ensures the generation of optimized LAMP primer sets that comply with distance, Tm, and mutation constraints. The output consisted of LAMP primer sets that met all specified constraints. These sets were stored in a CSV file along with their corresponding penalty scores and detailed information about each primer. This enabled specificity filtering and further scoring.

Primer Design

The first step involves generating candidate primers for six core regions of the target DNA sequence: F3, B3, F2, B2, F1c, and B1c. Using the Primer3 tool, primers were designed for

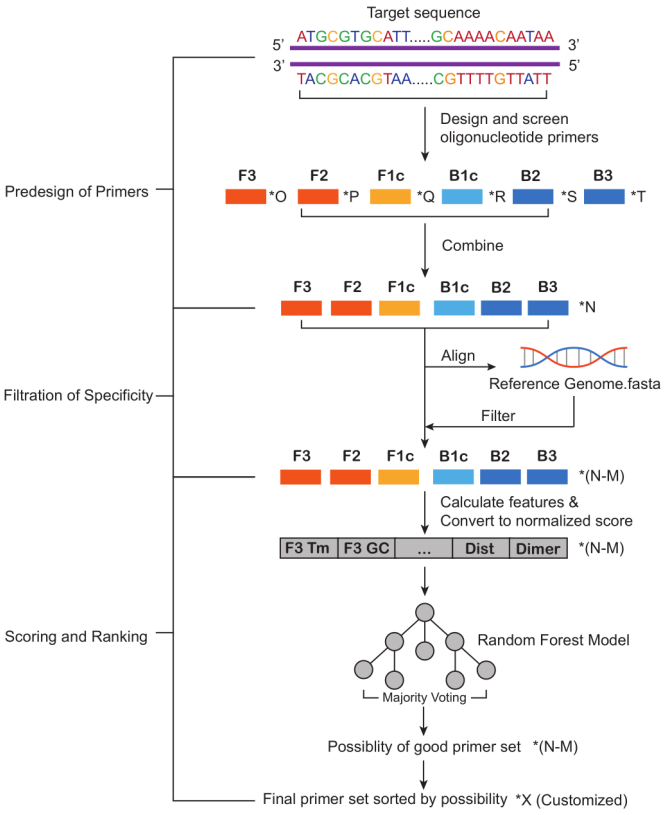

Fig. 2. **The workflow of LPAD.** Based on the input target sequences, the program performed primer predesign, specificity screening and scoring. It finally output the csv file of LAMP primer set ranked by scores.

the sense strand (F3, F2, B1c) and the antisense strand (B3, B2, F1c) (Untergasser et al. 2012). For each candidate primer, information such as position, length, Tm, and penalty scores was recorded.

To account for evolutionary variability, the algorithm also processed quasi-species sequences. Mutation frequencies at each position were calculated by comparing the main target sequence to the quasi-species sequences. Primers overlapping mutation-prone regions were penalized or removed to ensure stability and specificity. Finally, primers were sorted by position and penalty score, and those with excessive overlaps were filtered out based on a pre-defined maximum overlap threshold.

Primer Combination

After generating individual primers, a nested loop algorithm was employed to identify the optimal combination of LAMP primers for both ends of the DNA sequence. This process comprised the following steps:

1) **End-side Primer Combination:** Candidate primers were divided into forward (F3, F2, F1c) and reverse (B3, B2, B1c) sets. For each inner primer (F2 or B2), middle and outer primers (F3, B3) were iteratively combined, ensuring that combinations satisfied spacing constraints, such as minimum and maximum distances between primers. A penalty score was calculated for each combination, which included individual primer penalties, penalties for deviations in inter-primer distances, and overlap penalties. Combinations with scores exceeding a certain threshold were discarded. Additionally, loop primers (LoopF and LoopB) were designed and included in the scoring process. These primers were assessed for compatibility with the inner, middle, and outer primers to enhance amplification efficiency, with constraints such as minimum gaps and distance penalties applied to ensure optimal spacing and placement.

2) **Scoring and Final Combination:** The algorithm computed a composite score for each forward and reverse primer set. This score included penalties for Tm differences across primer pairs (e.g., between F1c and B1c), inter-primer distances, and individual primer penalties. Forward and reverse sets were then combined using an additional scoring function that accounted for the overall Tm balance and distance between the two sets. The resulting LAMP primer sets were ranked, and only those with the lowest penalty scores were selected for the final output.

B. Filtration of Specificity

To ensure the designed LAMP primers are specific to the target DNA sequence and do not bind non-target regions, we implemented a BLAST-based filtration module. This module involves constructing a BLAST database from the background genome or transcriptome, enabling efficient alignment of candidate primers (Altschul et al. 1997). The primers generated during the predesign step were aligned against the database using short-sequence alignment settings to assess their binding specificity. The alignment results were parsed to identify primers with off-target binding, and primers that aligned exclusively to the target region were retained. Non-specific primers, identified based on 100% identity and complete overlap with non-target regions, were excluded. The final output was a refined set of primers with high specificity to the target DNA sequence, saved in a CSV file for further processing and ranking.

C. Scoring and Ranking

To objectively evaluate and rank the filtered LAMP primer sets, we developed a scoring system based on primer features and a machine learning model. Each primer was evaluated based on key features such as GC content, melting temperature (Tm), free energy (Delta G), and hairpin structure, along with inter-primer distances and the likelihood of primer dimer formation. These features were then used to train a Random Forest classifier, which was trained on labeled data of valid and nonsense primers (Breiman 2001). The model predicted the probability of each primer set being high-quality based on the extracted features. The final ranking of primer sets was determined by combining the predicted probabilities and additional feature-based scores. This scoring and ranking system produced a ranked list of primer sets, prioritized by their suitability for LAMP assays. The final output enables researchers to select the most promising primer sets for experimental validation.

III. Reflection

Through collaborative development, we successfully created a fully automated tool for designing LAMP primers. This tool generates a scored and ranked primer set by taking a target sequence file and a background file as input. Before development began, our team precisely divided the problem into three modules: predesign, filtration, and scoring, and assigned

tasks accordingly, demonstrating our strong problem-analysis skills. Among these, Yihuan Xu and I were responsible for developing the primer predesign algorithm. We pre-defined the input and output formats for each module to ensure that the algorithms could be efficiently integrated during later stages. During development, each team member focused on their respective module's algorithm design. This approach allowed each component to be thoroughly addressed, refined, and tested. Once all modules were completed, we collaborated to integrate them into a unified program. At this stage, I designed the program architecture and integrated the three modules into a cohesive framework. Based on this, I worked with other team members to further optimize the program structure and conduct extensive testing, including evaluating environmental flexibility, input scalability, computational resource efficiency, and result reliability. In particular, we designated a member to collect necessary data to assist with benchmarking. Although the program's outputs do not completely align with those of mainstream LAMP tools, it successfully passed all conducted tests. This achievement reflects the robustness of our design and development process.

The development of the primer predesign algorithm formed the core of the program and involved complex calculations. At the early stage of the project, I developed an algorithm that divided the process into two parts: (1) independently designing four pairs of primers, and (2) combining primers based on linear distance constraints to identify potential LAMP primer sets. While this initial method showed potential, it encountered significant challenges. Primer pairs exhibited a tendency to cluster around specific "hotspots" in the sequence, limiting the quality and diversity required for optimal primer combinations (Fig 3). Furthermore, designing primer pairs was computationally expensive, with high time complexity that made the approach inefficient and difficult to scale.

After discussions with Yihuan, we realized that designing paired primers was not the most suitable approach for LAMP, particularly because the interaction of primers in LAMP differs significantly from that in traditional PCR. Reflecting on this, we decided to adopt the strategy used in LAVA, transitioning from paired primer design to the independent design of oligonucleotide primers (Torres et al. 2011). This change reduced computational resource usage and improved scalability. By designing independent oligonucleotide primers, we enabled free combination of primers, leading to a more diverse set of potential combinations. Unlike the previous approach, where only two primers within a pair could be combined, this method made it easier to satisfy subsequent linear distance constraints.

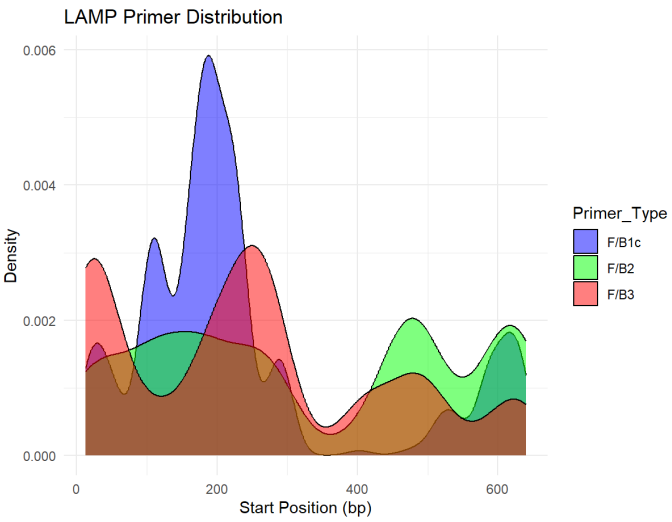

Fig. 3. **LAMP primer distribution.** The figure illustrates the distribution of start positions for three types of primers designed in pairs: inner primers (F/B1c), middle primers (F/B2), and outer primers (F/B3). Each curve represents the density estimation of primer start positions across the genome or targeted region, with different colors indicating the distinct primer sets.

Fig. 4. **Time resources occupied by primer predesign and specificity filtration.** (a) Predesign time consumption, illustrating the linear relationship between sequence length (bp) and time (s) required for primer predesign. (B) Specificity filtration time consumption, illustrating the linear relationship between the number of primer sets and the time (s) required for specificity filtration. The lines represent the fitted models of the two modules based on the tested data points, respectively. Both graphs highlight the computational efficiency of the LAMP primer design algorithm at different stages of the LPAD.

Given the high computational demands of the subsequent specificity filtering module, we recognized the need to minimize the number of primers passed to this step. This prompted us to introduce additional constraint criteria to reduce the number of primers requiring further testing. To address this, Yihuan and I collaboratively developed a penalty scoring system, applying stricter thresholds based on primer distances and Tm differences. This scoring mechanism streamlined the selection process and enhanced the specificity of the final primer sets. I was primarily responsible for designing the distance penalty model. Distance penalties were calculated using the custom function:

$$f(x) = a \cdot (b^x - 1),$$

where a is a scaling factor, $b > 1$ controls the growth rate, and x represents the deviations from ideal spacing. The exponential growth of penalties ensured that primers with larger spacing deviations were significantly penalized. While the default parameters a and b used in the function were not rigorously optimized, this approach effectively reduced the number of candidate primers. Additionally, quasi-sequence alignment and overlapping primer elimination were integrated to ensure primers would not bind to conserved regions and would avoid clustering.

Once the algorithm was completed, I focused on conducting performance testing. The module had not yet incorporated specificity filtering based on background files, making it suitable for benchmarking against PrimerExplorer5, which also lacks this feature. Testing revealed a high level of consistency between the primer sets generated by both approaches for the same target sequence Eiken Chemical Co., Ltd. (2019). However, upon examining the overlapping primer sets, we found that the primers designed by our algorithm often missed a few bases at both ends compared to those from PrimerExplorer5 (Table

I). It might due to differences between Primer3, which we use to produce oligonucleotide primers, and the internal algorithms employed by PrimerExplorer5. Additionally, we observed that some primer combinations only partially matched those from PrimerExplorer5. This may be attributed to our inclusion of loop primers, which introduced differences in constraint weighting during the combination process. To analyze the scalability of our algorithm, I generated random input sequences of varying lengths and evaluated runtime performance. The results showed that the time complexity grew linearly with sequence length, confirming the scalability of our method (Fig 4a). Although testing was limited due to time constraints and variability in sequence characteristics, the results demonstrated the feasibility and potential of our approach.

Tool	primer	location	sequence
PrimerExplorer	F1	357	atagttcggtttgaagatacgttga
LPAD	F1	355	ttatagttcggtttgaagatacg
PrimerExplorer	F2	293	ATCCAGATGGTTCTGTTCAA
LPAD	F2	300	TGGTTCTGTTCAATTGATCA
PrimerExplorer	F3	269	GTACTGGTGTAGACGTTGG
LPAD	F3	270	TACTGGTGTAGACGTTG
PrimerExplorer	B1	383	TAGACAAAGTAGCTCAAACATTGGC
LPAD	B1	383	TAGACAAAGTAGCTCAAACATTG
PrimerExplorer	B2	451	ttgtgaccattactgagttag
LPAD	B2	453	gtgaccattactgagttagt
PrimerExplorer	B3	483	tagagttagtgcacgtgt
LPAD	B3	475	taaggcaatagagttagtgc

TABLE I
COMPARISON OF RESULTS BETWEEN LPAD AND PRIMEREXPLORER5.

IV. Discussion

The development of the LPAD tool represents a significant step toward automating LAMP primer design, but there are several areas where the algorithms could be further improved. One key limitation of the current approach is the computational demand of the specificity filtering module, which requires BLAST-based alignment against large background genomes or transcriptomes (Fig 4b). While we introduced penalty-based filtering during the predesign stage to reduce the number of primers passed to this module, further optimization of the filtering thresholds or incorporating faster alignment tools could improve scalability. For instance, using tools like Bowtie or minimap2, which are optimized for speed and large-scale data, could replace BLAST for certain filtering tasks (Langmead 2009, Li 2018).

Another limitation is the reliance on default parameters in the scoring functions, such as the constants a and b in the distance penalty function. While these values were effective in reducing candidate primers, a more rigorous approach to determining optimal parameter values, perhaps through mathematical derivations or machine learning-based tuning, could further enhance the accuracy of primer scoring. Additionally, the predesign stage did not fully consider the specific characteristics of FIP and BIP primers. Refining the design process to include these factors could improve the overall efficiency and accuracy of the tool.

Comparing LPAD to other publicly available tools, such as PrimerExplorer5 and LAVA, reveals both strengths and limitations. Unlike PrimerExplorer5, which does not consider loop primers by default, LPAD includes loop primers, resulting in a more comprehensive design process. However, this introduced additional constraints, meaning the weight distribution of each condition needed to be reconsidered. Compared to LAVA, which uses a similar strategy of independent oligonucleotide primer design, LPAD incorporates a stricter penalty system and mutation filtering, potentially leading to higher specificity. However, LPAD currently lacks the advanced parameter optimization and user-friendly visualization features available in some other tools.

One potential direction for future improvement is integrating machine learning models into the predesign. For instance, deep learning models could predict primer efficiency and specificity based on sequence features, potentially outperforming traditional penalty-based scoring. Additionally, LPAD could benefit from modular enhancements, such as enabling users to dynamically adjust thresholds or select between different alignment algorithms based on their specific requirements.

Despite these limitations, LPAD provides a solid foundation for LAMP primer design and has the potential to evolve into a widely used tool. Its ability to automate the design, filtration, and scoring processes reduces user dependence on manual adjustments and minimizes the risk of human error. With further development, including optimization for large-scale datasets and expanded feature sets, LPAD could become a competitive alternative to existing tools.

V. References

Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W. & Lipman, D. J. (1997), 'Gapped blast and psi-blast: a new generation of protein database search programs', *Nucleic Acids Research* **25**(17), 3389–3402.

Breiman, L. (2001), 'Random forests', *Machine Learning* **45**(1), 5–32.

Eiken Chemical Co., Ltd. (2019), 'PrimerExplorer V5', https://primerexplorer.jp/e/. [Online; Accessed: 2024/12/11].

Langmead, B. (2009), 'Fast and sensitive dna alignment with-bowtie 1.0', *Genome Biology* **10**(3), R25.

Li, H. (2018), 'Minimap2: pairwise alignment for nucleotide sequences', *Bioinformatics* **34**(18), 3094–3100.

Notomi, T., Mori, Y., Tomita, N. & Kanda, H. (2015), 'Loop-mediated isothermal amplification (lamp): principle, features, and future prospects', *Journal of Microbiology (Seoul, Korea)* **53**(1), 1–5.

Notomi, T., Okayama, H., Masubuchi, H., Yonekawa, T., Watanabe, K., Amino, N. & Hase, T. (2000), 'Loop-mediated isothermal amplification of dna', *Nucleic Acids Research* **28**(12), E63.

Soroka, M., Wasowicz, B. & Rymaszewska, A. (2021), 'Loop-mediated isothermal amplification (lamp): The better sibling of pcr?', *Cells* **10**(8), 1931.

Torres, C., Vitalis, E. A., Baker, B. R., Gardner, S. N., Torres, M. W. & Dzenitis, J. M. (2011), 'Lava: an open-source approach to designing lamp (loop-mediated isothermal amplification) dna signatures', *BMC Bioinformatics* **12**, 240.

Untergasser, A., Cutcutache, I., Koressaar, T., Ye, J., Faircloth, B. C., Remm, M. & Rozen, S. G. (2012), 'Primer3–new capabilities and interfaces', *Nucleic Acids Research* **40**(15), e115.

Wong, Y. P., Othman, S., Lau, Y. L., Radu, S. & Chee, H. Y. (2018), 'Loop-mediated isothermal amplification (lamp): a versatile technique for detection of micro-organisms', *Journal of Applied Microbiology* **124**(3), 626–643.