Biotecnia
Journal of biological and health sciences
http://biotecnia.unison.mx
Universidad de Sonora
ISSN: 1665-1456

Original Article

The Not-alike3 command pipeline finds genomic targets for the molecular diagnosis of diseases caused by pathogen microorganisms

La tubería de comandos Not-alike3 encuentra blancos genómicos para el diagnóstico molecular de enfermedades causadas por microorganismos patógenos

Javier Montalvo-Arredondo*[†1], Marco A. Juárez-Verdayes[†1], Erika Nohemi Rivas-Martínez[2]

† They contributed equally

[1] Molecular Bioengineering and Bioinformatics Laboratory, Departamento de Ciencias Básicas de la Universidad Autónoma Agraria Antonio Narro. Calzada Antonio Narro 1923, Saltillo Coahuila México, CP. 25315 Tel. 844110200.

[2] Departamento de Botánica de la Universidad Autónoma Agraria Antonio Narro. Calzada Antonio Narro 1923, Saltillo Coahuila México, CP. 25315 Tel. 844110200.

ABSTRACT

DNA-based molecular techniques are crucial for the precise identification of microorganisms. Despite their speed and sensitivity, specificity hinges on primer or probe design. The search for unique sequences in organisms' genomes has various applications such as microorganism identification. There are published bioinformatic tools that employ the concept of *in silico* genomic subtractive hybridization analysis to identify species-specific/unique regions within a genome of interest. However, they use deprecated tools and programming languages that are not currently used in bioinformatics, some of them are specialized or difficult to obtain because the repositories, where these tools were hosted, are not public or the web servers are currently down. To address these issues, we implemented the *in silico* genomic subtractive hybridization idea in a user-friendly, open-source, freely available, easy-to-install command pipeline application written in Python (called Not-alike3). We designed PCR primers over the sequence of unique regions identified in the genomes of two *Mucorales* species, *Rhizopus oryzae* and *Cunninghamella bertholletia*, employing Not-alike3 command pipeline; we made specificity tests to challenge these primers and we observed that they were species-specific.

Keywords: Python programming, molecular diagnostics, pathogen detection, molecular biology, SHG.

RESUMEN

Las técnicas moleculares basadas en ADN son cruciales para la identificación correcta de microorganismos. A pesar de su velocidad y sensibilidad, la especificidad depende en el diseño de la sonda o el primer de PCR. La búsqueda por secuencias únicas en gnomas de organismos tiene varias aplicaciones como la misma identificación de microorganismos. Existen herramientas bioinformáticas publicadas que emplean el concepto de hibridación genómica sustractiva *in silico* para la identificación de regiones únicas o específicas de especie dentro de un genoma de interés. Sin embargo, éstas usan herramientas desactualizadas y/o lenguajes de programación que actualmente no son usados en el área de la bioinformática, otras herramientas son muy especializadas o son difíciles de obtener debido a que los repositorios (acervos) donde se hospedan las herramientas no son públicas o están apagados actualmente. Para abordar estos problemas, implementamos el concepto de hibridación genómica sustractiva *in silico* en una tubería de comandos escrita en Python que es amigable, de código abierto, gratis y fácil de instalar (llamada Not-alike3). Adicionalmente, diseñamos primers de PCR utilizando como molde la secuencia de regiones únicas identificadas en los genomas de dos especies de *Mucorales*, *Rhizopus oryzae* y *Cunninghamella bertholletiae* empleando la tubería Not-alike3, después realizamos evaluaciones de especificidad para desafiar esos primers que diseñamos y observamos que dichos primers fueron específicos de especie.

Palabras clave: programación Python, diagnóstico molecular, detección de patógenos, biología molecular, SHG.

INTRODUCTION

Traditional diagnostic methods, such as microbial culture, biochemical tests, hemagglutination inhibition tests, and enzyme-linked immunosorbent assays (ELISAs), are commonly employed. Among these techniques, pathogenic microbial culture is notably time-consuming, relying on morphological characteristics for identification, with associated issues of low specificity and sensitivity (Liu *et al.*, 2023). On the other hand, DNA-based molecular techniques such as PCR, quantitative PCR, fluorescent in situ hybridization (FISH), and biosensors among others, are useful methods to detect and identify microorganisms in biological samples, which are sensitive and fast, but specificity only relies on the primer or probe design (Vidic *et al.*, 2017). Species-specific primers or probes are designed on regions that are only present in the target genome. Some strategies to design highly specific primers involve the searching for species-specific genes that are unique in those species and whose sequences are used as templates to design specific primers. For example, *COTH* family are genes only present in Mucorales fungi and their sequences are used to identify these fungi at the order level, but in some cases are unable to discern between the closest related species belonging to the same genus (Baldin *et al.*, 2018). Other strategies use the variable regions of ribosomal genes ITS1, ITS2 or 28S D1/D2 domains (Voigt *et al.*, 1999; Wang *et al.*, 2014).

*Author for correspondence: Javier Montalvo-Arredondo
e-mail: buitrejma@gmail.com
Received: November 2, 2024
Accepted: April 27, 2025
Published: May 30, 2025

DOI: 10.18633/biotecnia.v27.2495

One of the species-specific sequences isolated from human genome was HS5, carried out by a technique called "genomi subtraction" were genomic DNAs from two related species, in this case human and non-human hominid, are blended in an in vitro competitive reassociation. If a sequence of human genome region is highly similar to a sequence in non-hominid genome, these DNA strands reassociate in vitro to form a hybrid after a denaturation process, which are unable to amplify by PCR. If a human genomic region is highly divergent between human and non-human genome, DNA strands are going to reassociate with themselves, and these non-hybrid DNAs can be amplified. This technique allows the isolation of species-specific genomic sequences known as unique regions (Ueda *et al.*, 1990).

These unique regions can be used for multiple purposes such as templates to design species-specific primers and enhance the specificity of DNA-based techniques to detect and identify microorganisms in biological samples. The growth of biological sequences databases alongside the evolution of computer processors allowed to perform *in silico* subtractive genomic hybridization with the same goal, to find unique regions in a genome of interest.

In silico subtractive hybridization analysis is a powerful conceptual tool, but the available computer applications use deprecated tools or programming languages that are not commonly used in bioinformatics, and in some cases, the commands pipelines are just protocols and not a computer application or are specialized just for bacterial genomes, and some of them are difficult to obtain because the repositories are not public or the web servers are currently down (Argimón *et al.*, 2014; Barh *et al.*, 2011; Chetouani *et al.*, 2001; Haubold *et al.*, 2021; Portela *et al.*, 2010; Shao *et al.*, 2010; Singh y Mishra, 2010). Haubold *et al.* (2021), have developed a new tool that uses a similar basis for the same purpose, but it is programmed in a language that is not popular in bioinformatics. Thus it will be difficult to maintain by bioinformatics community.

To address these issues, we wrote a user-friendly, open-source, freely-available, easy-to-install command pipeline computer application in Python language, a popular programming language in bioinformatics, which is packed with Steuptools and Wheel packages and is installed alongside its Python packages dependencies with Pip Installs Packages. This commands pipeline is called Not-alike3 and uses the *in silico* genomic subtractive hybridization idea in an iterative whole-genome sequence comparison, between a genome of interest and a set of different genomes, to find unique genomic regions and use them as templates to design species-specific primers and enhance the specificity of DNA-based molecular techniques to identify the living being of interest. In this paper, we present the commands workflow of this computer application and the results of the specificity test for PCR primers designed over unique regions of genomes of two Mucorales species, identified by Not-alike3.

MATERIALS AND METHODS

Software requirements and description

To run this pipeline, the following bioinformatics software is required, Blast+ (Camacho *et al.*, 2009), Stringtie (Pertea *et al.*, 2016), Hisat2 (Kim *et al.*, 2019), Samtools (Li *et al.*, 2009), Gffread (Pertea y Pertea, 2020), NCBI dataformat and datasets (O'Leary *et al.*, 2024), Primer3_core (Untergasseer *et al.*, 2012), and the following Python packages OS, Subprocess, Click, Biopython, Ctypes and Pandas which, some of them, can be easily downloaded from PYPI server (https://pypi.org) or they are already installed for Python version 3.10 or higher. The source code for this project is freely available at https://doi.org/10.5281/zenodo.10557734 or https://github.com/exseivier/not-alike-3.0.0.

Mucorals species unique regions identification

To identify these dissimilar unique regions in *Rhizopus oryzae* and *Cunninghamella bertholletiae*, we searched a genome database depicted in Supplementary table 1 and used the following searching strategy, window size: 1000 bp, step size: 250 bp, percentage of identity: 25 %, percentage of query HSP coverage 12 %, expected value 100, gap open, gap extension, match, mismatch were set as default, the Blast task was "blastn". Primers were designed over these unique regions with search-primers command with the following parameters: primer length: 20 nt, Tm: 58.5 °C, GC percentage: 60 %, fragment size: 100-500 bp. Two primer pairs with the best quality scores for each *Mucorales* species *R. oryzae* (RO_135 and RO_290) and *C. bertholletiae* (CB_160 and CB_348) were selected (Table 1).

DNA extraction

Six *Mucorales* strains (*R. oryzae*, *C. bertholletiae*, *Rhizopus delemar*, *Mucor plumbeus*, and 2 strains of *Lichtheimia ramosa*), stored at -75 °C in freezer stocks, were grown at saturation on YPD-agar plates. After growing, a sample of the biomass was taken with a toothtip and resuspended in 200 µL of 10 mM Tris-HCl and 1 mM EDTA pH 7.5 solution (TE), then 200 µL of lysis buffer with 2 % CTAB dissolved in TE were added and gently mixed. Protein separation was conducted using 400 µL of Clororphorm – Isoamyl alcohol 25:1. Samples were vigorously mixed using a vortex for 1 min and centrifuged at 13000 rpm for 10 min. Aqueous phase was taken for DNA precipitation using chilled ethanol, by adding 40 µL of sodium acetate 3 M (pH 5) and 1 mL of 96 % ethanol. Samples were gently mixed and centrifuged at 13000 rpm for 10 min. The supernatant was discarded and the DNA pellet was washed with 1 mL of 70 % ethanol.

Species specificity test

To test the specificity of these primers, we performed PCR experiments with blended genomic DNA samples of several Mucorales species available in our laboratory: *R. oryzae*, *C. bertholletiae*, *Rhizopus delemar*, *Mucor plumbeus*, and 2 strains of *Lichtheimia ramosa*. We prepared several blended samples with equal amounts of genomic DNA of every *Mucorales* spe-

Table 1. PCR primers designed over unique regions of *R. oryzae* and *C. bertholletiae* genomes.
Tabla 1. Primers de PCR disñados en regions únicas de los genomas de *R. oryzae* y *C. bertholltiae*.

Name	Species	Sequence	Length (nt)	Tm (°C)	Pair name	Size
RO_135_F	*R. oryzae*	AGACCAAAGTCAGGACGGC	19	59.6	RO_135	135bp
RO_135_R	*R. oryzae*	TTTTCTCACGCATGCGGC	18	58.5		
RO_290_F	*R. oryzae*	TTTCCAAATGCCGCAGCC	18	58.8	RO_290	290bp
RO_290_R	*R. oryzae*	AGCACGTTGTCCTTGAGGG	19	59.7		
CB_160_F	*C. bertholletiae*	AGGGATTACTTCTCGCGCG	19	59.1	CB_160	160bp
CB_160_R	*C. bertholletiae*	TGTCCTCGAAGCCAAACCG	19	60.4		
CB_348_F	*C. bertholletiae*	GCTCCATCGCTTTTTCCGG	19	59.4	CB_348	348bp
CB_348_R	*C. bertholletiae*	AAGCAGTGACAGTACCGCC	19	58.9		

cies and then primer pairs RO_135 (fwd and rev) with CB_348 (fwd and rev) were tested together or alone, performing respectively a multiplex or conventional PCR, also, we tested the primer pairs RO_290 (fwd and rev) with CB_160 (fwd and rev) in the same way. As a control, we used universal 28S (D1/D2 domains) primers (Voigt *et al*., 1999; Wang *et al*., 2014).

In another experiment we prepared 4 samples, the first sample contained equal amounts of genomic DNA of all *Mucorales* species, in the second sample we blended the genomic DNA of Mucorales except the genomic DNA of *R. oryzae*, in the third sample we excluded the genomic DNA of *C. bertholletiae* and in the fourth sample we excluded both genomic DNAs. With these samples, we also tested primer pairs (RO_290 and CB_160) and (RO_290 and CB_348) in a similar PCR experiment. In all cases, we used the universal 28S primer pair as a control as we did in the previous PCR experiment.

PCR reactions were done with 2X Crystal Master Mix Taq Polymerase (Jena Bioscience), we added the primers oligonucleotide at 0.4 µM. Thermocycling program was as follows: initial denaturing step at 95 °C for 5 min, 35 cycles of [denaturing at 95 °C for 30 sec, annealing at 59 °C for 30 sec and extension at 72 °C for 1 min], and a final extension step at 75 °C for 10 min. Samples were stored at 18 °C. Gel electrophoresis was done with agarose 1.5 %.

RESULTS AND DISCUSSION

Software requirements and description

Not-alike3 is a commands pipeline written mainly in Python language (version > = 3.10) that uses several open-source free-available bioinformatics tools, described in materials and methods, and identifies unique regions of a genome of interest comparing its sequence with sequences from genome databases of related species and employs a procedure called *In silico* Genomic Subtractive Hybridization. Not-alike3 contains several commands namely, db-makeblast, db-makefile, search, serach-primers, show-db, show-exp and assm-stats. The db-makeblast and db-makefile commands are used to build the genomes BLAST database used in the searching procedure, the search command executes the command pipeline that identifies the unique regions, the search-primers command designs primers over the identified unique dissimilar regions. Command pipeline communication between Python and GNU/Linux Bash was implemented with functions of the Subprocess Python package. There are other miscellaneous commands, show-db, show-exp and assm-stats, the first and second commands show information about the data base and the parameters used in previous searching tasks, and the assm-stats command calculates the assembly statistics. Not-alike3 has a command line user interface built with the Click Python package that permits the user to choose the command, and pass the arguments needed (see usage information in Figure 1).

The searching for unique regions and the designing of primers is divided into three main tasks (1) BLAST database building, executed just one time when a new database is required or the existing one is modified, (2) the searching for unique regions which is the iterative whole-genome sequence comparison to subtract those unique regions found in the genome of interest, and (3) primers designing which uses the unique sequences as templates. Not-alike3 was successfully tested in computers with 3 GHz AMD RYZEN3 and 2.5 GHz Intel Celeron processors with 12 Gb and 4 Gb RAM respectively.

First task: BLAST database building

Not-alike3 is compatible with genome packages downloaded from NCBI datasets (https://www.ncbi.nlm.nih.gov/datasets/). We use datasets and dataformat NCBI command line tools to manipulate genome data packages employing

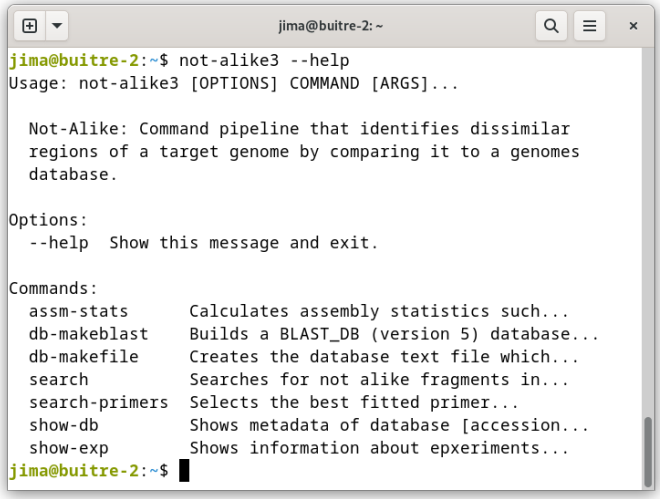

Figure 1. Not-alike3 usage information. It shows the available commands and the main description.
Figura 1. Información de uso de Not-alike3. Muestra los commandos disponibles y la dscripción general.

the wrapping commands db-makeblast and db-makefile. This enables working with an extensive and diverse array of genomes. To build the BLAST database, Not-alike3 contains a command called db-makeblast that takes the JSON file that contains the NCBI dataset metadata. This command finds the location where FASTA files are and formats them in BLAST DB files (version 5) with makeblastdb (Camacho *et al.*, 2009). The command db-makefile makes use of the JSON metadata file to create a text file with the paths and names of BLAST DB files, this text file is subsequently used in the search for unique regions. The db-makefile command executes a genome database sorting procedure based on a quick sequence similarity analysis to sort genomes from the most similar to the most dissimilar one compared with the genome of interest. To accomplish that, the sequence of the genome of interest is split into fragments of 500 nucleotides each 250 nucleotides using a sliding-window algorithm, then this command takes a random sample representing 10 % of the total number of fragments. These sampled fragments are used to query database genomes, in a genome-by-genome manner, with Blastn (Camacho *et al.*, 2009). Then, every genome is sorted based on the number of obtained hits, from the higher to the lower number (see usage information in Figure 2).

Second task: The searching for unique regions

Not-alike3 contains a command called search that performs the searching for unique regions, every searching task originates a process identifier PID to track the results. In this command, we implemented the idea of *in silico* genomic subtractive hybridization by an iterative genome-by-genome sequence comparison analysis. This command executes three main steps. In the first step, as input, it employs a FASTA file that contains the sequence of the genome of interest and splits it into fragments of a size and at a step size determined by the user, using a sliding-window algorithm. In some cases, we handle DNA sequences in Python with Biopython package functions.

However, to perform a subtle analysis, it is recommended to split the genome at a small step size. In an extreme example, the sequence can be split into fragments of determined size with a step size equal to one, so if each split subsequence is stored in Biopython's DNA sequence object, the total fragments will occupy a measurable amount of space in RAM. To represent these fragments in a memory-efficient way, we implemented in C language a dynamic arrays' data structure called "DNA" that resembles DNAStringSet of R Biostring package (Pagés *et al.*, 2024), where fragments sequences are represented in a single nucleotide sequence which is the entire genome in a character array, and the information to track each fragment sequence is stored in several dynamic arrays. We used the Ctypes Python package to write wrapper functions to handle C data structure and functions from the shared library (libdna.so) which is in the Not-alike3 package.

In the second step, Not-alike3 performs an iterative search using these fragments stored in the FASTA file as input to query each one of the database genomes with Blastn (Camacho *et al.*, 2009), in a genome-by-genome manner. In the first search, the fragments that align at any sub-sequence in the first queried genome of the database are eliminated from the fragments FASTA file, maintaining only the query fragments that did not hit any subsequence. This task is known as the filtering and header updating procedure and is repeated until the last genome of the database is queried (Figure 3). To perform this procedure, the headers of FASTA sequences that hit any subsequence of the queried genome are stored in a linked list C data structure that was used to search into the "DNA" data structure. When the header of the fragment in DNA structure is found, the hide tag is set to one and the header which is in a node of the linked list is deleted, the memory is free, and the previous node is linked to the

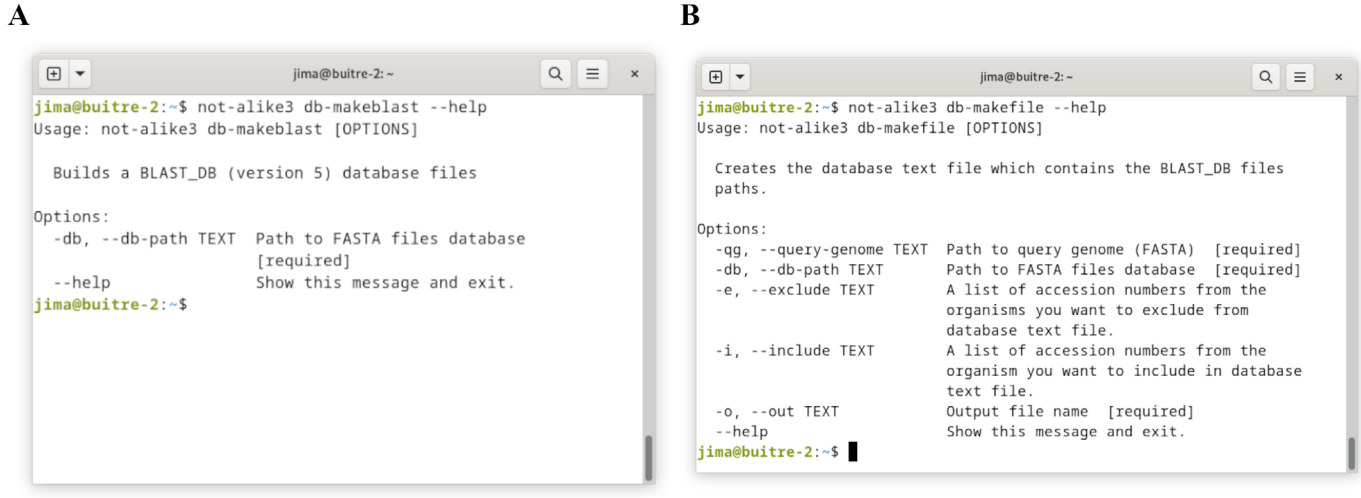

Figure 2. BLAST-database building commands usage information. It shows the usage information for (A) db-makeblast command and (B) db-makefile command.

Figura 2. Comandos para la construcción de la base de datos BLAST e información de uso. Muestra la información de uso para los comandos (A) db-makeblast y (B) db-makefile.

next node. We decided to use a linked list data structure to facilitate data modification of the hits list obtained from Blastn output.

Using a sorted database from the most similar to the most dissimilar genome compared to the genome of interest, using the iterative search with the elimination of fragments that hit a subsequence in the queried genome, increases the computing speed in each iteration. This is because in the first searching procedure, a high proportion of fragments will be eliminated from FASTA file, and the time of execution of the subsequent Blastn searching will be less compared with the previous iteration.

In the third step, the remained fragments that did not hit any subsequence of database genomes are assembled using a genome-guided procedure: first, Not-alike3 maps fragments to reference genome and handles alignment BAM files with Hisat2 (Kim *et al.*, 2019) and Samtools (Li *et al.*, 2009), second, it assembles mapped fragments with Stringtie (Pertea *et al.*, 2016) with the parameters set as default throughout the entire procedure. The sequences of the assembled fragments are stored in a FASTA file and the genome coordinates of those assembled fragments are transformed and stored in a GTF file employing Gffreads (Pertea y Pertea, 2020) tool to visualize them in genome browsers. These output files are stored in the output gtfs folder and these assembled fragments represent the unique regions. The search command takes several arguments: the genome FASTA file name, genome database path, and TOML configuration file name. Other arguments are passed inside the TOML file. An example of a TOML configuration file is described in (Supplementary Figure 1).

Third task: The primers designing

The search-primers command is used to design primers over these unique genomic regions. This command uses Primer3_core (Untergasseer *et al.*, 2012) to design primers. The search-primers command uses the unique regions FASTA file as input which is obtained as a result of the search command execution, to create Primer3 input file and to design the primers. After primer designing, this command sorts the primer sequences by the quality score. Arguments are passed by command line options and the requirements are shown in (Figure 4). Search-primers returns a text file (extension *.sort.prm) with the designed primer pairs sorted by quality score and stores it into the output gtfs folder. There are other miscellaneous commands called assm-stats, show-db and show-exp. The command assm-stats calculates the assembly statistics. The command show-db prints on screen the information about the genome database, which includes the genomes accession number, name of species and taxon ID. The show-exp command shows information about parameters used in previous searching tasks, executed inside the folder where the output "log" folder is.

Illustrative examples

To get more information, please visit the following video tutorial by clicking the link: https://youtu.be/rwltheAmX0Y?si=78GIpzH-C6T6a7uL

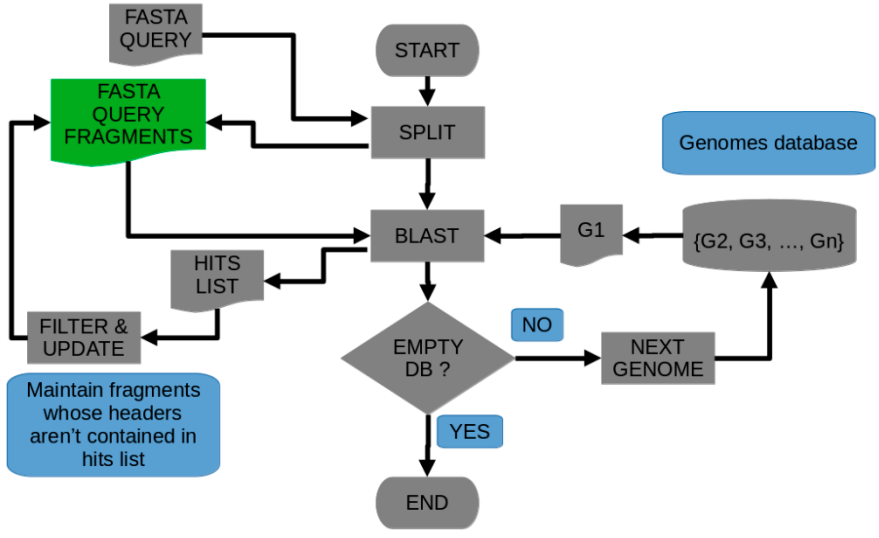

Figure 3. Not-alike3 search-command flowchart. This figure shows the split and iterative Blastn searching procedures querying genomes from database. Each time a searching is performed, the FASTA QUERY FRAGMENTS file is updated (FILTER & UPDATE procedure; a.k.a. filtering and header updating) dropping out those fragments whose headers were contained in the Blastn hits list.

Figura 3. Diagrama de flujo del commando "search" de Not-alike3. Esta figura muestra los procedimientos de división de secuencias y búsqueda itrativa por Blastn consultando las bases de datos de los genomas. Cada ocasión que una búsqueda es realizada, el archivo FASTA "QUERY FRAGMENTS" es actualizado con el procedimiento FILTER & UPDATE (actualización de cabeceras y filtrados) descartando aquellos fragmentos cuyas cabeceras fueron contenidas en la lista de los hallazgos de Blastn.

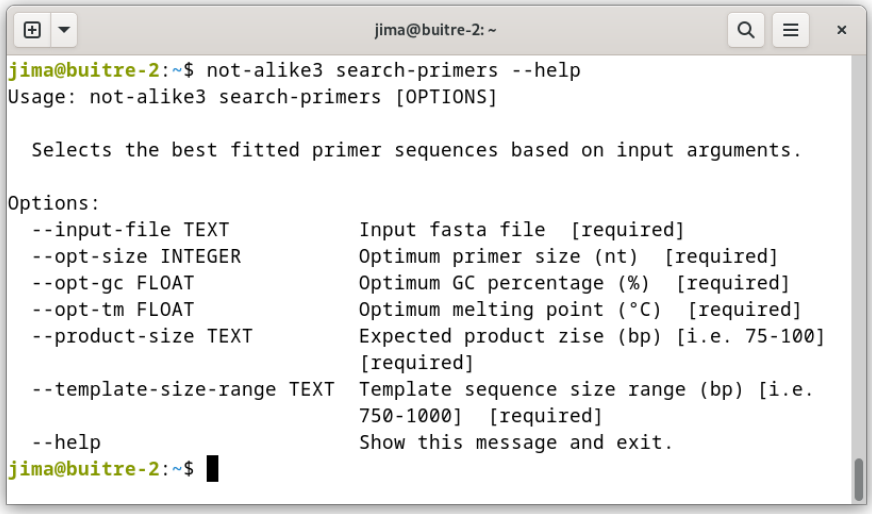

Figure 4. Search-primers command usage information. It shows the required arguments and the description of the command.
Figura 4. Información de uso del comando "search-primers". Muestra los argumentos requeridos y la descripción del comando.

Potential use

In this paper, we present a potential use of Not-alike3 in molecular diagnostics for pathogen detection. We describe the results of specificity tests of Polymerase Chain Reaction (PCR) primers designed over unique regions identified with Not-alike3 in two *Mucorales* species genomes of *Rhizopus oryzae* (GCA_000149305.1) and *Cunninghamella bertholletiae*(GCA_000697215.1). These primers (see Table 1) were challenged with blended samples of genomic DNA of several *Mucorales* species.

Almost all primer pairs amplified the expected band size (contrast Figure 5 and Table 1). In (Figure 5A) the results of PCR experiments are shown when a mix of genomic DNA of *Mucorales* was tested with primer pairs (RO_135 and CB_348) and (RO_290 and CB_348). We observed that RO_290 (lanes six and seven, 290 bp), CB_160 (lanes six and eight 160 bp) and CB_348 (lanes three and five, 348 bp) primer pairs amplified a fragment of the expected size and RO_135 (lanes three and four, 135 bp) pair did not amplify any band. We eliminated the RO_135 primers pair in the next PCR experiments.

In (Figure 5B) show results from PCR experiments where we prepared 4 different blended, in one sample we mixed the genomic DNA of all *Mucorales*, in the other three samples we mixed almost all genomic DNAs of all *Mucorales*, but we excluded *R. oryzae* or *C. bertholletiae* or both genomic DNAs, and then we challenged these samples with primers pairs (RO_290 and CB_160). We observed the two expected bands when both genomic DNAs of *R. oryzae* and *C. bertholletia* were present in the sample (lane three). Also, we observed the expected band sizes only when the genomic DNA of *C. bertholletiae* (lane five) or *R. oryzae* (lane seven) was present in the mixed sample. No band was observed when both genomic DNAs were absent in the sample (lane nine). Lanes two, four, six and eight are controls where we used universal primers (NL1 and NL4) that amplify a region of 28S rRNA gene (approximately 750 bp).

We also tested the primer pairs (RO_290 and CB_348) in a similar PCR experiment and similar results were observed as shown in Figure 5C. In this figure the two expected bands when both genomic DNAs of *R. oryzae* and *C. bertholletiae* were present in the sample can be observed (lane three). We observed the specific band for *C. bertholletiae* (lane five) and *R. oryzae* (lane seven). No bands were observed when *R. oryzae* and *C. bertholletiae* genomic DNAs were absent in the blended sample. Also, in this experiment, we used the same universal primers as controls (28S, lanes two, four, six, and eight).

We showed in these series of experiments that Not-alike3 can find genomic targets that are species-specific and can be exploited in molecular techniques such as PCR for diagnosis of diseases caused by pathogen microorganisms. Nevertheless, Not-alike3 could be used to identify species-specific genomic regions to use them in other molecular techniques such as, real-time PCR, quantitative PCR, digital PCR, Fluorescent *in situ* Hibridization (FISH) and DNA- or RNA-based biosensors.

Limitations and future work

The reliability of this bioinformatics approach relies on the available database information. Not-alike3 command pipeline requires at least some DNA or RNA sequence information for the target microorganism (e.g. pathogen microorganism) to perform the comparative analysis with a group of genomes of organisms that are known to proliferate together in the same niche. Thus, the specificity in the design of the primers or probes could be compromised by database information lacking.

Assembling errors of the target genome can cause technical artifacts during sequence comparisons. These errors

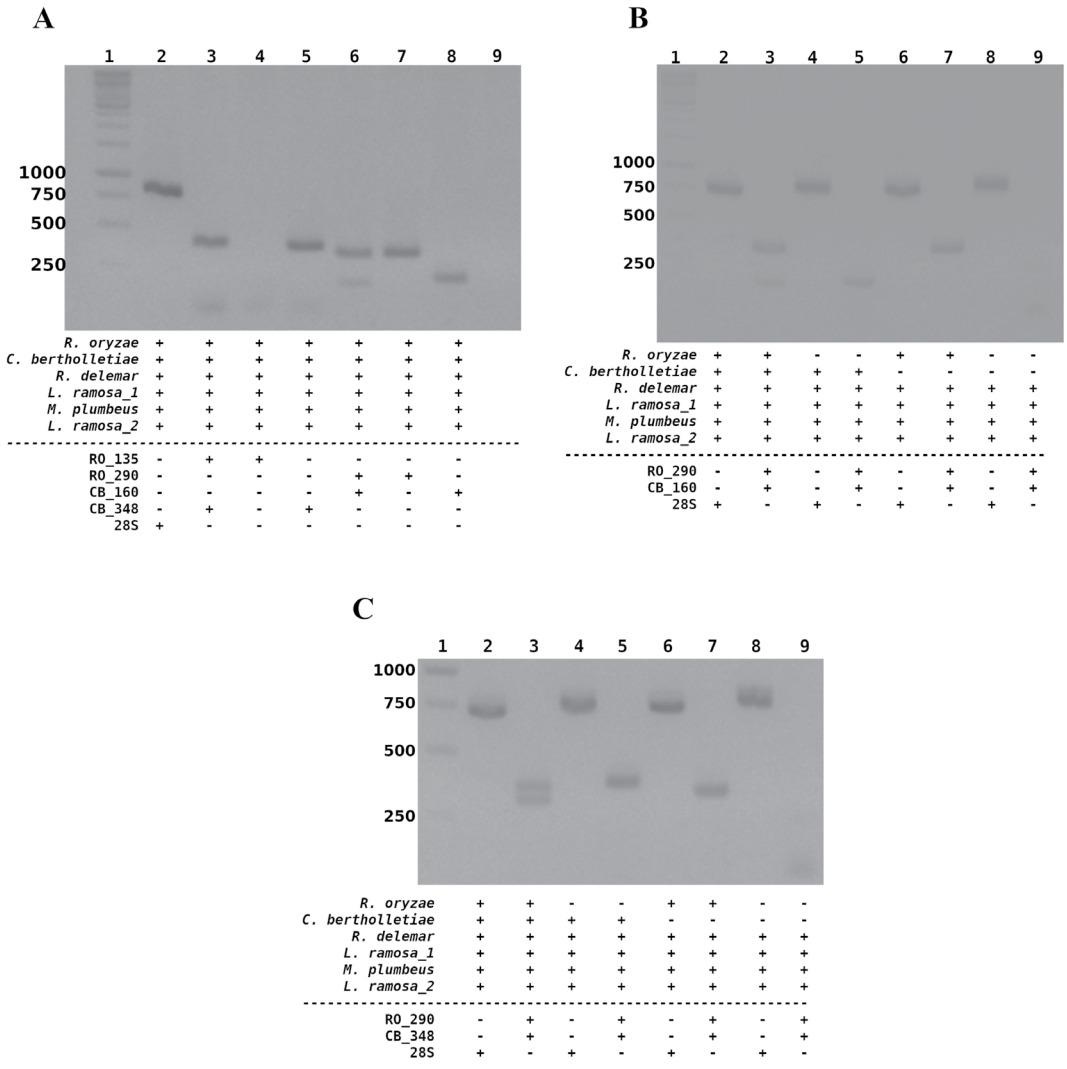

Figure 5. PCR primers specificity test. (**A**) Specificity test for primers pairs RO_135, RO_290, CB_160 and CB_348 over mixed samples of equal amounts of genomic DNA of *Mucorales* species. Lanes two to eight are samples of blended genomic DNA of all six *Mucorales* strains challenged with 28S primers (lane two), RO_135 and CB_348 primers (lane three), RO_135 primers (lane four), CB_348 (lane five), RO_290 and CB_160 primers (lane six), RO_290 primers (lane seven), CB_160 primers (lane eight) and no primers (lane nine). (**B**) Specificity test for primers pairs RO_290 and CB_160 over mixed samples of equal amounts of genomic DNA of *Mucorales* species where genomic DNA of *R. oryzae* or *C. bertholletiae* were absent or not. Samples of blended genomic DNA for all *Mucorales* strains (lanes two and three), blended genomic DNA samples lacking *R. oryzae* genomic DNA (lanes four and five), blended genomic DNA samples lacking *C. bertholletiae* genomic DNA (lanes six and seven), blended genomic DNA samples lacking both *R. oryzae* and *C. bertholletiae*. Those samples were challenged with: 28S primers (lanes two, four, six and eight), RO_290 and CB_160 (lanes three, five, seven and nine). (**C**) Specificity test for primers pairs RO_290 and CB_348 over mixed samples of equal amounts of genomic DNA of *Mucorales* species where genomic DNA of *R. oryzae* or *C. bertholletiae* were absent or not, and the blended genomic DNA samples were organized as shown in section B of this figure, but samples were challenged with: 28S primers (lanes two, four, six and eight), RO_290 and CB_348 primers (lanes three, five, seven and nine). For every subsection in this figure, lane one is a 1kb plus ladder (ThermoFisher), and the band sizes are measured in base pairs.

Figura 5. Prueba de especificidad de los cebadores de PCR. (A) Prueba de especificidad para los pares de cebadores RO_135, RO_290, CB_160 y CB_348 sobre muestras mixtas de cantidades iguales de ADN genómico de especies de Mucorales. Los carriles dos a ocho son muestras de ADN genómico mezclado de las seis cepas de *Mucorales* desafiadas con cebadores 28S (carril dos), cebadores RO_135 y CB_348 (carril tres), cebadores RO_135 (carril cuatro), CB_348 (carril cinco), cebadores RO_290 y CB_160 (carril seis), cebadores RO_290 (carril siete), cebadores CB_160 (carril ocho) y sin cebadores (carril nueve). (B) Prueba de especificidad para pares de cebadores RO_290 y CB_160 sobre muestras mixtas de cantidades iguales de ADN genómico de especies de *Mucorales* donde el ADN genómico de *R. oryzae* o *C. bertholletiae* estaba ausente o no. Muestras de ADN genómico mezclado para todas las cepas de *Mucorales* (carriles dos y tres), muestras de ADN genómico mezclado que carecían del ADN genómico de *R. oryzae* (carriles cuatro y cinco), muestras de ADN genómico mezclado que carecían del ADN genómico de *C. bertholletiae* (carriles seis y siete), muestras de ADN genómico mezclado que carecían tanto de *R. oryzae* como de *C. bertholletiae*. Esas muestras fueron desafiadas con: cebadores 28S (carriles dos, cuatro, seis y ocho), RO_290 y CB_160 (carriles tres, cinco, siete y nueve). (C) Prueba de especificidad para pares de cebadores RO_290 y CB_348 sobre muestras mixtas de cantidades iguales de ADN genómico de especies de *Mucorales* donde el ADN genómico de *R. oryzae* o *C. bertholletiae* estaba ausente o no, las muestras de ADN genómico mezcladas se organizaron en los carriles como se muestra en la sección B de esta figura, pero las muestras se desafiaron con: cebadores 28S (carriles dos, cuatro, seis y ocho), cebadores RO_290 y CB_348 (carriles tres, cinco, siete y nueve). Para cada subsección en esta figura, el carril uno es una escalera de 1 kb plus (ThermoFisher), y los tamaños de banda se miden en pares de bases.

produce unreal sequences that could be identified as highly species-specific genomic regions by Not-alike3, as a result of a low probability of finding in the database similar sequences to unreal erroneous sequences, created by assembling errors, in the target genome. This scenario represents another limitation of this command pipeline. Because it is unable to identify the assembling errors of the target genome and the database, we recommend exploring the feasibility of two or more dissimilar regions identified by Not-alike3.

One of the important tasks that Not-alike3 performs is the sequence comparison. In this software version, we implemented Blastn, a program that queries a genome database to find similar sequences to a query sequence, to accomplish this task. Although this program is memory efficient, it could be computationally expensive if the size of the database is huge which is the case of Not-alike3 analysis. Despite the heuristic algorithm we implemented during the querying of databases to improve the processing of sequence comparisons, we think that this represents an area of improvement that could be addressed with new technology such as alignment-free based sequence comparison methods or artificial intelligence.

CONCLUSIONS

If primer or probe design can discern a difference between a genome of interest among other genomes of related species, by sequence similarity during primer or probe annealing, thus the DNA-based molecular technique becomes highly specific with the ability to detect a microorganism of interest in a sample with plenty microorganisms without the need of isolating it (Davi *et al.*, 2021; Gardés *et al.*, 2012; vanWeezep *et al.*, 2019). Consequently, the need to find unique regions in the genome of interest arises, to use these unique regions as templates to design species-specific primers or probes for their use in DNA-based molecular techniques.

For a current software alternative, we have developed a user-friendly, open-source, freely-available and easy-to-install commands pipeline named Not-alike3. This pipeline, written in the Python language, is designed to run efficiently on low-capacity PC machines. It is conveniently packed with Setuptools and Wheel, to ease the installation, including all necessary dependencies, through Python Installs Packages (pip).

We identified unique regions in the genomes of two *Mucorales* species employing Not-alike3, and the primers designed using these unique regions as templates showed to be species-specific. Moreover, these primer designs could be adapted for multiplex PCR to detect two or more unique sequences in the same sample. In consequence, we think this command pipeline has potential applications in molecular diagnostics for pathogen detection and identification.

SUPPLEMNTARY FILES

The following supporting information can be downloaded at **Figshare: Supplementary Table 1**. Mucorales genome database used in Not-alike3 analysis. (xlsx and ods files) DOI: https://doi.org/10.6084/m9.figshare.24463867.

Supplementary Figure 1. Configuration file and database text file DOI: https://doi.org/10.6084/m9.figshare.24463678.v1.

ABBREVIATIONS

nt.	Nucleotides.
PCR.	Polymerase Chain Reaction.
fwd.	Forward PCR primer.
rev.	Reverse PCR primer.
bp.	Base pairs.
DNA.	Deoxyribonucleic acid.
ITS1.	Internal Transcribed Spacer 1.
ITS2.	Internal Transcribed Spacer 2.
28S.	Large Subunit (LSU) of ribosomal DNA gene.

ACKNOWLEDGMENTS

We acknowledge "Coordinación de la División de Ingeniería" and "Departamento de Ciencias Básicas" of "Universidad Autónoma Agraria Antonio Narro" for the facility infrastructure we employed to develop and test Not-alike3.

CONFLICTS OF INTEREST

The authors declared no conflicts of interest.

REFERENCES

Argimón, S., Konganti, K., Chen, H., Alekseyenko, A.V., Brown, S. and Caufield, P.W. 2014. Comparative genomics of oral isolates of *Streptococcus mutans* by *in silico* genome subtraction does not reveal accessory DNA associated with severe early childhood caries. Infection, Genetics and Evolution. 21: 269–278. DOI: https://doi.org/10.1016/j.meegid.2013.11.003.

Baldin, C., Soliman, S.S., Jeon, H.H., Alkhazraji, S., Gebremariam, T., Gu, Y., Bruno, V.M., Cornely, O.A., Leather, H.L., Sugrue, M.W., Wingard, J.R., Stevens, D.A., Edwards, J.E. and Ibrahim, A.S. 2018. PCR-based approach targeting mucorales-specific gene family for diagnosis of mucormycosis. Journal of Clinical Microbiology. 56(10): 1110–1128. DOI: https://doi.org/10.1128/jcm.00746-18.

Barh, D., Tiwari, S., Jain, N., Ali, A., Santos, A.R., Misra, A.N., Azevedo, V. and Kumar, A. 2011. *In silico* subtractive genomics for target identification in human bacterial pathogens. Drug Development Research. 72(2): 162–177. DOI: https://doi.org/10.1002/ddr.20413.

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K. and Madden, T.L. 2009. BLAST+: architecture and applications. BMC bioinformatics. 10: 1–9. DOI: https://doi.org/10.1186/1471-2105-10-421.

Chetouani, F., Glaser, P. and Kunst, F. 2001. FindTarget: software for subtractive genome analysis. Microbiology. 147(10): 2643–2649. DOI: https://doi.org/10.1099/00221287-147-10-2643.

Davi, M.J.P., Jeronimo, S.M.B., Lima, J.P.M.S. and Lanza, D.C.F. 2021. Design and *in silico* validation of polymerase chain reaction primers to detect severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Scientific Reports. 11: 12565 https://doi.org/10.1038/s41598-021-91817-9.

Gardès, J., Croce, O. and Christen, R. 2012. *In silico* analyses of primers used to detect the pathogenicity genes of *Vibrio cholerae*. Microbes Environ. 27(3): 250–625. DOI: https://doi.org/10.1264/jsme2.me11317.

Haubold, B., Klötzl, F., Hellberg, L., Thompson, D. and Cavalar, M. 2021. Fur: Find unique genomic regions for diagnostic PCR. Bioinformatics. 37(15): 2081–2087. doi: https://doi.org/10.1093/bioinformatics/btab059.

Kim, D., Paggi, J.M., Park, C., Bennett, C. and Salzberg, S.L. 2019. Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. Nature biotechnology. 37(8): 907–915. DOI: https://doi.org/10.1038/s41587-019-0201-4.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R. and 1000 Genome Project Data Processing Subgroup. 2009. The sequence alignment/map format and SAMtools. Bioinformatics. 25(16): 2078–2079. DOI: https://doi.org/10.1093/bioinformatics/btp352.

Liu, Q., Jin, X., Cheng, J., Zhou, H., Zhang, Y. and Dai Y. 2023. Advances in the application of molecular diagnostic techniques for the detection of infectious disease pathogens (Review). Mol Med Rep. 27(5):104. DOI: https://doi.org/10.3892/mmr.2023.12991.

O'Leary, N.A., Cox, E., Holmes, J.B., Anderson, W.R., Falk, R., Hem, V., Tsuchiya, M.T.N., Schuler, G.D., Zhang, X., Torcivia, J., Ketter, A., Breen, L., Cothran, J., Bajwa, H., Tinne, J., Meric, P.A., Hlavina, W. and Schneider, V.A. 2024. Exploring and retrieving sequence and metadata for species across the tree of life with NCBI Datasets. Sci Data. 11(1): 732. DOI: https://doi.org/10.1038/s41597-024-03571-y

Pagès, H., Aboyoun, P., Gentleman, R. and DebRoy, S. 2024. Biostrings: Efficient manipulation of biological strings. Bioconductor R version package 2.70.2. DOI: https://doi.org/doi:10.18129/B9.bioc.Biostrings.

Pertea, G. and Pertea, M. 2020. GFF utilities: GffRead and GffCompare. F1000Research. 9. DOI: https://doi.org/10.12688%2Ff1000research.23297.2.

Pertea, M., Kim, D., Pertea, G.M., Leek, J.T. and Salzberg, S.L. 2016. Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown. Nature protocols. 11(9): 1650–1667. DOI: https://doi.org/10.1038/nprot.2016.095.

Portela, J., Grunau, C., Cosseau, C., Beltran, S., Dantec, C., Parrinello, H. and Boissier, J. 2010. Whole-genome in-silico subtractive hybridization (WISH)-using massive sequencing for the identification of unique and repetitive sex-specific sequences: the example of *Schistosoma mansoni*. BMC genomics. 11: 1–8. DOI: https://doi.org/10.1186/1471-2164-11-387.

Shao, Y., He, X., Harrison, E.M., Tai, C., Ou, H.Y., Rajakumar, K. and Deng, Z. 2010. mGenomeSubtractor: a web-based tool for parallel *in silico* subtractive hybridization analysis of multiple bacterial genomes. Nucleic Acids Research. 38: W194–W200. DOI: https://doi.org/10.1093/nar/gkq326.

Singh, V. and Mishra, R.K. 2010. RISCI-Repeat Induced Sequence Changes Identifier: a comprehensive, comparative genomics-based, *in silico* subtractive hybridization pipeline to identify repeat induced sequence changes in closely related genomes. BMC bioinformatics. 11: 1–25. DOI: https://doi.org/10.1186/1471-2105-11-609.

Ueda, S., Washio, K. and Kurosaki, K. 1990. Human-specific sequences: Isolation of species-specific DNA regions by genome subtraction. Genomics. 8(1): 7–12. DOI: https://doi.org/10.1016/0888-7543(90)90219-K.

Untergasser, A., Cutcutache, I., Koressaar, T., Ye, J., Faircloth, B.C., Remm, M. and Rozen, S.G. 2012. Primer3 – New capabilities and interfaces, Nucleic Acids Research. 40(15): e115–e115. DOI: https://doi.org/10.1093/nar/gks596.

van Weezep, E., Kooi, E.A. and van Rijn, P.A. 2019. PCR diagnostics: *In silico* validation by an automated tool using freely available software programs. J Virol. Methods. 270: 106–112. DOI: https://doi.org/10.1016/j.jviromet.2019.05.002.

Vidic, J., Manzano, M., Chang, C.M. and Jaffrezic-Renault, N. 2017. Advanced biosensors for detection of pathogens related to livestock and poultry. Veterinary Research. 48(1): 1–22. DOI: https://doi.org/10.1186/s13567-017-0418-5.

Voigt, K., Cigelnik, E. and O'donnell, K. 1999. Phylogeny y PCR identification of clinically important *Zygomycetes* based on nuclear ribosomal-DNA sequence data. Journal of Clinical Microbiology. 37(12): 3957–3964. DOI: https://doi.org/10.1128/jcm.37.12.3957-3964.1999.

Wang, X., Fu, Y.F., Wang, R.Y., Li, L., Cao, Y.H., Chen, Y.Q., Zhao, H.Z., Zhang, Q.Q., Wu, J.Q., Weng, X.H., Cheng, X.J. and Zhu, L.P. 2014. Identification of clinically relevant fungi and prototheca species by rRNA gene sequencing and multilocus PCR coupled with electrospray ionization mass spectrometry. PLoS One. 9(5): e98110. DOI: https://doi.org/10.1371/journal.pone.0098110