Funding Scheme: THEME [ICT-2007.8.0] [FET Open]

Paving the Way for Future Emerging DNA-based Technologies: Computer-Aided Design and Manufacturing of DNA libraries

Grant Agreement number: 265505

Project acronym: CADMAD

Deliverable number: D2.8

Deliverable name: SW for efficient and intelligent design of complex DNA editing

Contractual Date[1] of Delivery to the CEC: M24
Actual Date of Delivery to the CEC: M25
Author(s)[2]: Tuval ben Yehezkel
Participant(s)[3]:WEIZMANN
Work Package: wp2
Security[4]: Pub
Nature[5]: R
Version[6]: 0.0
Total number of pages:

1 As specified in Annex I
2 i.e. name of the person(s) responsible for the preparation of the document
3 Short name of partner(s) responsible for the deliverable
4 The Technical Annex of the project provides a list of deliverables to be submitted, with the following classification level:
Pub - Public document; No restrictions on access; may be given freely to any interested party or published openly on the web, provided the author and source are mentioned and the content is not altered.
Rest - Restricted circulation list (including Commission Project Officer). This circulation list will be designated in agreement with the source project. May not be given to persons or bodies not listed.
Int - Internal circulation within project (and Commission Project Officer). The deliverable cannot be disclosed to any third party outside the project.
5 **R (Report):** the deliverables consists in a document reporting the results of interest.
P (Prototype): the deliverable is actually consisting in a physical prototype**,** whose location and functionalities are described in the submitted document (however, the actual deliverable must be available for inspection and/or audit in the indicated place)
D (Demonstrator): the deliverable is a software program, a device or a physical set-up aimed to demonstrate a concept and described in the submitted document (however, the actual deliverable must be available for inspection and/or audit in the indicated place)
O (Other): the deliverable described in the submitted document can not be classified as one of the above (e.g. specification, tools, tests, etc.)
6 Two digits separated by a dot:
The first digit is 0 for draft, 1 for project approved document, 2 or more for further revisions (e.g. in case of non acceptance by the Commission) requiring explicit approval by the project itself;
The second digit is a number indicating minor changes to the document not requiring an explicit approval by the project.

Abstract

CADMAD's library problem covers a huge space of possible solutions. Various stringological data structures and algorithms were harnessed to aid with represent, compress, optimizing and construct the user designed libraries.

Keywords[7]**:**

DAWG, LCS

Like many complex problems, the problem of designing the editing steps, comprising a construction tree for CADMAD's libraries can be met with a divide and concur approach. The resulting sub-problems are tackled by integrating various point optimization algorithms, the pairing algorithm, the backend connecting system and the automation.
We will present the planning's workflow from DNALD to a construction ready plan as well as a through, step-by-step explanation of the process applied on one of the CADMAD partner's libraries and will analyse the possible savings in promised within the various optimizations.

1. Implementation

- User designed library's sequences set is compressed to an optimal DAWG with the naturally emerging DNA references as the alphabet in use.
- The DAWG's synthetic fragments are extracted into a separate, naïve DAWG, with natural DNA bases as the alphabet.
  - Resulting shared sub-sequences within the synthetic sequences are noted.
  - Additional shared fragments within the synthetics are identified using LCS and enrich the DAWG.
  - Short nodes are compressed together and possibly inserted into neighbouring primers in order to reduce the cost of the required synthetic DNA the number of DNA assembly steps.
- Synthetics DAWG is reincorporated to the full library's DAWG.
- The pairing algorithm reinflates the library's DAWG back to its explicit-sequences form, while documenting the order of fragments concatenations as a construction tree.

[7] Keywords that would serve as search label for information retrieval

2. Results

Panke's library:

DNALD description:

```
inputs {
             pSEVA261 := 2519 base pairs long NF
             pSEVA231 := 3112 base pairs long NF
             pSEVA281 :=  2179 base pairs long NF
             pSEVA271 := 2136 base pairs long NF
             pSEVA291 := 3122 base pairs long NF
             pSEVAfba := 4059 base pairs long NF
       }

       pUC    :=pSEVA261[263:1207]
       pBBR1  :=pSEVA231[263:1800]
       pMB1   :=pSEVA281[263:867]
       p15A   :=pSEVA271[263:824]
       pSC101 :=pSEVA291[263:1548]

       FBA    :=pSEVAfba[10:1100]

       T1     :=pSEVA231[143:262]


       P1     :=
'ctggttttccagcagacgacggagcaaaaactacccgtaggtgtagttggcgcaagcgtccgattagctcaggttttaagatg'
       P2     :=
'tttccagcagacgacggagcaaaaactacccgtaggtgtagttggcgcaagcgtccgattagctcaggttttaagatg'
       P3     :=
'agcagacgacggagcaaaaactacccgtaggtgtagttggcgcaagcgtccgattagctcaggttttaagatg'
       P4     := 'cgacggagcaaaaactacccgtaggtgtagttggcgcaagcgtccgattagctcaggttttaagatg'
       P5     := 'agcaaaaactacccgtaggtgtagttggcgcaagcgtccgattagctcaggttttaagatg'

       R1     := 'gacaaaaatctagaaataattttgtttaactttaagaaggagatatacaa'
       R2     := 'gggagctaacgagggcaaaaa'
       R3     := 'aataattttgtttaactttaagaaggagatatacat'
       R4     := 'atggtgttctccaatttttattaaattagtcgctacgagatttaagacgt'
       R5     := 'ctctaaaagcgcgctgaaacaagggcaggtttccctgccctgtgattttt'

outputs {
       ^Library := (pSC101 + p15A + pMB1 + pBBR1 + pUC) T1 (P1 + P2 + P3 + P4 + P5) (R1 +
       R2 + R3 + R4 + R5) FBA
  }
```

Graphical description (as DAWG):

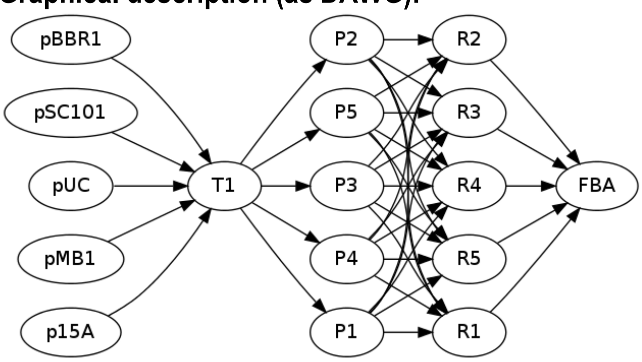

Figure A: Panke user design DAWG

The concatenation of the P1-5 synthetics set with the R1-5 synthetics set arouse from the user's design but has no physical justification since both sides are synthetic. Hence, the library's graph is equivalent to the following graph:

Simply concatenating the adjacent synthetic fragments implied by the user design:

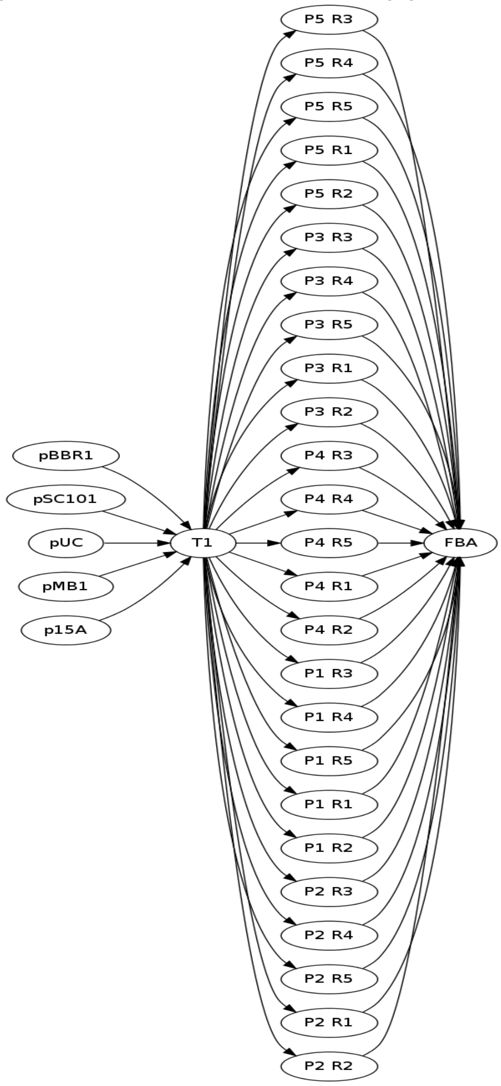

Figure B: Panke concatenated synthetics DAWG

Once concatenated, we scan the full synthetic fragments for shared substrings and logically compress the synthetics subgraph to a minimal DAWG:

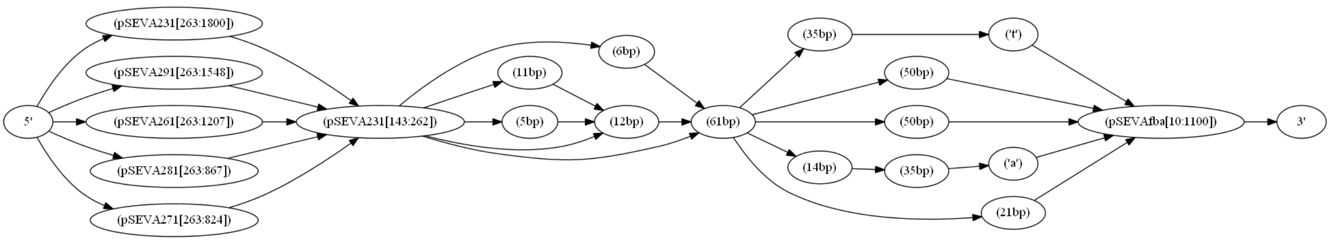

Figure C: Panke optimized DAWG

In this example, we can see a shared synthetic fragment that was discovered regardless of the user's design and will be very helpful in terms of compressing the construction tree.

Cost assessments:
In this example the user design enforces two construction options regarding the synthetic nodes of the library. The straight forward option will be to order synthetic fragments as designed (figure1), with the addition of required 25bp long overlapping sequences on each side, leading to two groups of 25 oligos each with lengths ranging from 72bp to 125bp. Overall 50 oligos, averaging around 83bp.
The slightly less straight forward option will be to treat the concatenated set of synthetic nodes (figure1) as the oligos, ordering the 25 nodes with the addition of 25bp long overlapping sequences on each side, leading to a single group of 25 oligos, ranging from 133bp to 185bp and averaging around 166bp. Hence saving a construction step and perhaps a full generation (construction tree level).

After our algorithmic optimization of the DAWG, we realize that a 58bp long fragment is shared throughout the synthetics set and could potentially be synthetized just once. When zooming on the optimized synthetics graph we can easily see that post-optimization, we can construct the full synthetics combinatorial set with eleven or less oligos, not longer than those proposed by the straight forward construction plan, therefor slicing the amount of required oligos and biochemical concatenations in more than half.

Figure D: Panke optimized synthetics only DAWG

Many of the synthetic fragments resulting from this optimization algorithm are short enough to be incorporated within the primers that assist the concatenation of their adjacent neighbours, saving oligos, biochemical operations and most importantly reducing the depth of the construction tree.

3. Conclusions

We've shown that coupling the DAWG and iterative LCS can drastically reduce a library's synthetic fragments cost in all its aspects, oligos length, oligos number and number of resulting concatenations required.
The source code of this software is available for review.**References**

4. Abbreviations

List all abbreviations used in the document arranged alphabetically.

DAWG	**Directed Acyclic Words Graph**
LCS	**Longest common substring (implemented using a suffix tree)**
Stringology	**Mathematical logic/theoretical computer science area that deals with string processing**