ε-다중목적 진화연산을 이용한 DNA Microarray Probe 설계

조영민° 신수용 이인희 장병탁
서울대학교 컴퓨터공학부
{ymcho°, syshin, ihlee, btzhang}@bi.snu.ac.kr

A Probe Design Method for DNA Microarrays Using ε-Multiobjetive Evolutionary Algorithms

Youngmin Cho°, Soo-Yong Shin, In-Hee Lee, and Byoung-Tak Zhang
School of Computer Science and Engineering, Seoul National University

요 약

최근의 생물학적인 연구에 DNA microarray가 널리 쓰이고 있기 때문에, 이러한 DNA microarray를 구성하는데 필요한 probe design 작업의 중요성이 점차 커져가고 있다. 이 논문에서는 probe design 문제를 thermodynamic fitness function이 2개인 multi-objective optimization 작업으로 변환한 뒤, ε-multiobjective evolutionary algorithm을 이용하여 probe set을 찾는다. 또한, probe 탐색공간의 크기를 줄이기 위하여 각 DNA sequence의 primer 영역을 찾는 작업을 진행하며, 사용자가 직접 프로그램을 테스트할 수 있는 웹사이트를 제공한다. 실험 대상으로는 mycoides를 선택하였으며, 이 논문에서 제안된 방법을 사용하여 성공적으로 probe set을 발견할 수 있었다.

1. 서 론

DNA oligonucleotide microarray는 probe라는 DNA sequence로 구성되어 있으며, probe는 관심대상이 되는 gene의 complementary 형태로 만들어서, 이에 결합반응을 보이는 해당 gene을 파악할 수 있도록 도와준다. 특히 DNA microarray는 한번의 실험으로 각 gene의 probe 결합을 동시에 표현해주기 때문에 생물학 연구에서 그 활용도가 점차 높아지고 있으며, 이러한 DNA microarray의 quality와 직결되는 probe 설계의 문제도 생물정보학의 중요한 연구분야로 자리잡고 있다.

좋은 probe를 판단하는 기준은 여러가지가 있지만[1], 가장 핵심적인 내용은 target gene으로부터 complementary substring을 추출하여 만든 probe가 non-target gene과의 반응을 최소화한채 target gene에만 안정적으로 반응하도록 하는 것이다. 이를 위하여 다양한 probe 설계 기법들이 제시되어 왔는데, 전통적으로는 생물학적인 parameter를 이용하여 결합 free energy를 계산하거나 BLAST 등으로 sequence를 정렬시키는 방법 등이 선호되었고, 최근에는 machine learning 기술을 도입하여 성능을 향상시키려는 시도도 나타나고 있다.

이 논문에서는 기계학습의 최적화 기법 중 하나인 multi-objective evolutionary algorithms(MOEA)를 핵심으로 probe를 설계하였으며, 동시에 이 과정의 전후에 성능보강을 위한 몇가지 방안을 도입하였다. MOEA는 상충할 수 있는 목표를 동시에 추구해야 하는 문제에서 convergence와 diversity를 함께 만족시키는 solution 집합을 찾는 진화연산이다[2]. 이번 연구에서는 기존에 사용한 ε-MOEA[1]를 두개의 objective를 가지는 형태로 구현하였으며, 특히 계산 시간이나 코드의 안정성 측면에서도 큰 개선을 이루어냈다.

또한 이렇게 probe를 찾는 진화연산 과정의 search space에 해당하는 primer region을 추정하는 루틴을 preprocessing 형태로 포함해서 진화연산의 방향에 긍정적인 영향을 미치도록 하였다. 그리고, melting temperature나 cross hybridization 기준에 의해 ε-MOEA의 결과물인 Pareto-optimal set을 재평가한뒤 그 중 한 개의 set을 추천하는 부분을 포함시켜서 probe 설계의 다양한 기준을 고루 충족시키고자 하였다.

앞으로 2장에서는 구현한 probe 설계 알고리듬을 단계별로 나누어서 구체적으로 기술하며, 3장에서는 실험대상인 mycoides를 이용한 실험 결과를 평가하여 2장의 기법이 어떤 효과를 지니고 있는지를 설명한다. 4장에서는 향후 과제를 비롯한 토의사항을 제시한다.

2. Probe 설계

2.1 Primer 영역 탐색

Primer는 PCR(Polymerase Chain Reaction) 과정에서 중요하게 여겨지는 DNA fragment이며, multiple sequence를 정렬시킬때 일종의 기준점 역할을 하게 된다. 즉, left primer와 right primer의 쌍으로 존재하는 primer는 sequence들간에 유사도가 높은 부분에 해당하고, 반대로 primer pair 사이의, 낮은 유사도를 보이는 공간은 sequence를 잘 구별하도록 도와준다. 따라서, 각 sequence의 specificity가 높아지는 이러한 primer 사이의 영역에서 probe를 탐색한다면, 보다 효율적으로 결과를 얻을 수 있을것이다. 이번 연구에서는 [3]을 참조하여, 다음과 같은 방식으로 primer를 추정한다.

1) clustalw[4]를 사용하여 전체 gene sequence를 정렬시키고, 그 결과에 해당하는 consensus sequence를 추출한다. consensus sequence는 IUPAC ambiguity code를 이용하여 정렬결과를 표시한 것이며 consensus sequence에서 A, C, G, T 와 같이 기본 type으로 나타나는 부분은 각 sequence에서 공통으로 같은 위치에 존재한다는 뜻이므로 primer로 선택될 확률이 높다.

2) 각 sequence에 대해서 primer3[5]를 사용하여 최적 길이를 24bps, 최적 melting temperature를 65°C로 설정한 left, right primer를 각각 50개씩 추출한다. 추출한 primer는 1)의 consensus string과 비교하여 전체 sequence마다 비슷한 영역이 있는지를 간접적으로 검사하게 되는데, 이때의 유사도 기준으로는 base의 type이 일치하는 비율을 살펴보는 non-degeneracy ratio와, 비교 영역에서 gap이 얼마나 작은가를 살펴보는 non-gapped bps ratio를 사용한다.

3) 유사도 평가를 통과한 left primer와 right primer의 모든 조합을 고려하고 그 중에서 둘 사이의 영역 크기가 해당 sequence 전체 길이의 일정 비율 이상인 pair만을 고려하기로 한다. 그리고 이중에서 left primer와 right primer의 melting temperature 차이가 가장 작은 pair를 분석 대상이 된 sequence의 최종 primer pair로 선택한다.

이렇게 구한 primer 사이의 영역은 probe를 본격적으로 찾는 다음 단계인 MOEA의 search space에 해당하므로, 2)와 3)의 ratio를 어떻게 조정하느냐에 따라서 계산시간과 최종 probe의 quality가 영향을 받게 된다. 예를 들어, 세 parameter의 값이 커질수록 primer를 엄격한 기준에 의해서 선발하므로, 동시에 primer 사이의 공간 - MOEA의 search space - 도 점점 커지고 극단적으로는 MOEA 단계에서 primer없이 sequence의 전체 영역으로부터 probe를 찾게 된다. 즉, primer 영역에서 얻어지는 probe 탐색 guidance를 어느 수준으로 활용할 것인가를 문제마다 적절하게 결정해야 한다.

2.2 ε-MOEA를 이용한 Pareto-optimal set 생성

MOEA는 기존의 진화연산이 사실상 한개만의 fitness function을 가진 문제에 적용되어 최적의 solution 한개를 구하는데 목적이 있는 것에 반하여, 때로는 상충되기도 하는 여러 objective를 동시에 최적화한뒤 solution의 집합 형태로 최종 답을 제시하는 machine learning 기법이다. 이러한 진화과정에서 solution들간의 우열을 판별해주는 개념이 domination인데, X, Y를 solution, f_i를 i번째 fitness function 이라고 할때, 아래와 같은 조건이 만족되면 X가 Y를 dominate한다고 정의한다. (maximization 기준)

$$^{\forall}i,\ f_i(X) \geq f_i(Y)$$
$$^{\exists}i,\ f_i(X) > f_i(Y)$$

따라서, domination 관계는, 하나의 fitness function이라는 유일한 기준으로 모든 solution 정렬이 가능했던 일반적인 진화연산과 달리, 다른 solution에 dominate 당하지 않는 non-domination 이라는 개념을 도입하고 이를 만족하는 solution을 집합 형태로 제공한다. 그리고, probe 설계 문제 역시 아래와 같이 나열할 수 있는 다양한 objective를 가지고 있기 때문에 MOEA를 이용한 접근이 바람직하다[1].

1) probe는 non-target gene과 결합하지 않아야 한다.
2) probe는 secondary structure를 구성하지 않아야 한다.
3) probe는 melting temperature가 균일해야 한다.

실제 구현에서는 MOEA의 결과로 얻은 Pareto-optimal set으로부터 한개의 solution만을 선택할때의 기준 중 하나로 3)을 사용하였으며(2.3 참조), MOEA에서는 1), 2)를 목표로 설정한 후 mfold[6]로 측정한 sequence간의 free energy로 그 값을 표현하였다. 즉, 1)은 sequence와 probe 사이의 결합을 방지해야 하므로 둘간의 free energy를 최대화 하는 방향으로, 2) 역시 probe 자체가 스스로 결합하지 않도록 그때의 free energy를 최대화하는 방향으로 진화하도록 설정하였다.

구체적으로 target gene의 벡터를 $T = (t_1, t_2, t_3, \cdots, t_n)$, 이에 대응하는 probe의 벡터를 $P = (p_1, p_2, p_3, \cdots, p_m)$, 각 probe를 $p_i = \{A, C, G, T\}^l$ (l은 probe의 length) 라고 하면, 1)과 2)는 다음과 같은 수식으로 표현 가능하다[1]. 참고로, 진화과정의 기본 연산인 crossover는 대상이 되는 두 probe 벡터 P_1, P_2에 대하여 p_{ij}와 p_{2j}를 선택적으로 교환하도록 만들어져 있으며, mutation은 probe가 target gene sequence의 특정 위치로부터 추출한 substring을 complementary하게 처리해서 만들어지는 것에 착안하여, probe 추출위치를 1 bp 만큼 shift 시킨뒤 새로운 probe를 만드는 방법으로 구현하였다.

$$\text{maximize} : f_1(P) = \sum_{i \neq j} \text{free energy}(p_i, t_j)$$
$$\text{maximize} : f_2(P) = \sum_{i} \text{free energy}(p_i)$$

알고리듬의 측면에서 보면, 이번 논문에서는 현재 연구되고 있는 다양한 형태의 MOEA중에서 ε-MOEA를 사용하였다. ε-MOEA는 ε-dominance 관계를 이용하여 solution을 평가하고 elite solution을 archive 에 저장하면서 진화하는 steady-state 알고리듬이다[2]. ε-dominance는 아래의 수식처럼 dominance의 개념이 정의되는 공간을 ε라는 상수 크기만큼의 block으로 나눈뒤, solution이 속한 block 간의 dominance를 가지고 solution을 비교하는 개념인데, 특히 ε-MOEA는 한 block안에 한개의 solution만을 유지하기 때문에 사용자가 정의하는 ε에 따라서 적절한 convergence와 diversity를 동시에 얻을 수 있다.

$$^{\forall}i,\ \lceil f_i(X) / \varepsilon \rceil \geq \lceil f_i(Y) / \varepsilon \rceil$$
$$^{\exists}i,\ \lceil f_i(X) / \varepsilon \rceil > \lceil f_i(Y) / \varepsilon \rceil$$

ε-MOEA의 전체적인 진행방식은 아래와 같다. 참고로, 이번에 구현한 프로그램은 이전의 연구[1]에서 사용한 버전의 버그를 수정하고 성능을 개선한 것이며, 압도적으로 많은 시간을 소비하는 free energy 계산루틴의 결과를 중간 저장하는 자료구조를 도입하여서 수행시간을 90%이상 감소시켰다.

1) 초기 population 랜덤생성
2) non-dominated individual을 archive로 복사
3) population과 archive로부터 부모를 선택하여 individual 생성
   a) population에서 2개의 individual 랜덤선택
   b) dominate하는 individual 선택 (관계가 없으면 랜덤선택)
   c) archive에서 1개의 individual 랜덤선택
   d) 정해진 rate에 따라서 crossover, mutation 수행
4) archive 갱신
   a) 새로운 individual에 의해 ε-dominate 되는 기존멤버 교체
   b) 두 individual이 같은 grid에 있다면 경계에 가까운 것 보존
   c) 기존 individual과 ε-domination 관계가 없다면 새롭게 영입
5) population 갱신
   a) 새로운 individual에 의해 dominate 되는 기존멤버 교체
   b) 기존멤버와 domination 관계가 없다면
      랜덤선택한 individual 교체
6) 예정된 generation에 도달할때까지 3)~5)를 반복

2.3 최종 probe 선정

2.2의 ε-MOEA 과정을 통하여 Pareto-optimal set인 probe 벡터의 집합을 얻은 뒤에는, 사용자가 필요한 한개의 probe 벡터를 추천해야한다. 이 논문에서는 두가지 기준에 의해 각각의 추천 solution을 선정하는데, 첫번째는 앞에서 언급한 probe와 target-gene이 반응하는 melting temperature(Tm)의 균일성이다. 이 작업은 mfold[6]를 이용하여 Tm을 계산하는 방식으로 이루어졌으며, 결과의 표준편차가 작을수록 실제 실험환경에서 안정적으로 동작한다고 간주할 수 있다.

다른 하나의 기준은 probe와 non-target gene 사이에서 발생하는 cross-hybridization의 정도이다. 이는 2.2에서 mfold로 구한 free energy와 유사하지만, BLAT[7]을 일종의 checker program으로 이용하는 형태이며 따라서 hybridization block count가 제일 작은 probe 벡터를 선택한다. 결국, 이러한 solution filtering 작업은, MOEA로부터 얻을 수 있는 결과 집합의 신뢰성에 다양한 평가기준을 유연하게 결합시켜주며, 3장의 실험에도 활용되었다.

2.4 Web interface

사용자가 직접 테스트할 수 있도록 2장에서 기술된 내용이 http://cbit.snu.ac.kr/~probe/eMOEA.php에 구현되어 있으며, 제출한 gene에 대한 probe 선정 결과는 이메일로 전송된다.

3. 실험

3.1 데이터

우폐역을 일으키는 mycoides gene sequence 17개를 한양대학교 APR lab으로부터 제공받아서 최상의 결과를 찾아보았고, gene의 길이는 309 ~ 1513 bp 이며 진화연산의 기본설정은 crossover rate = 0.9, mutation rate = 0.01, population = 10으로 정하였다.

3.2 결과

그림 1은 primer 정보를 배제하고 population이 70일 때, 진화가 진행되면서 60부근의 blat search count라는 답을 찾아가는 패턴을 보인다. 또한, Tm의 표준편차는 이번 MOEA의 objective에서 배제했으므로 별다른 성과를 거둘 수 없음을 확인하였다.

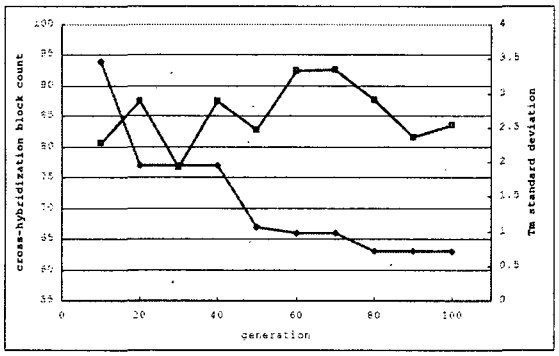

그림 1. generation에 따른 성능비교.

그림 2는 primer region 선정 결과에 초점을 맞춘 것으로써, non-degeneracy ratio의 영향을 드러낸다. 즉; ratio가 높으면 MOEA에서 더 넓은 영역을 탐색하므로 동일한 계산비용 기준으로 성능이 하락하는, convergence 의 문제점이 드러나고, 반대로 ratio가 낮으면 좁은 영역에서만 solution을 찾기 때문에 diversity가 부족해져서 역시 성능이 하락하는 것을 알 수 있다.

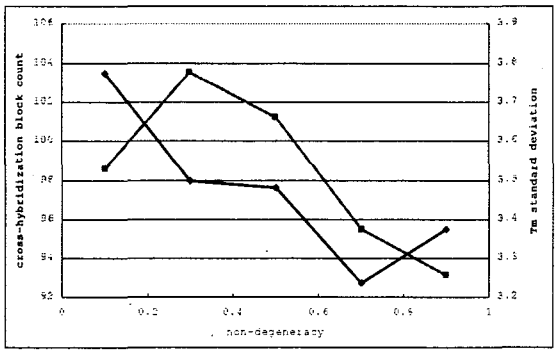

그림 2. primer region 선정에 따른 성능비교.

그림 3은 MOEA의 ε 선정 결과가 미치는 영향이며, ε을 크게하면 덜 정밀한 기준으로 진화하기 때문에 결과로 얻는 답의 수는 줄어들지만, 최적화하고자 하는 대상에 따라서는 성능을 유지하면서 적은 계산 비용을 들이게 된다. 즉, 2.2의 기준 1)에 대해 ε을 1에서 10으로 증가시켰지만, 성능이 유지되는 것을 볼 수 있다.

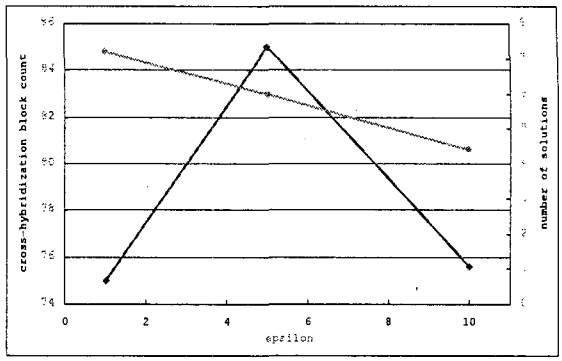

그림 3. ε-MOEA의 ε에 따른 성능비교.

4. 결론

본 논문에서는 mycoides gene의 probe를 찾는 작업을 여러개의 최적화 대상을 가진 문제로 변환한 뒤, 기계학습 관점에서 ε-MOEA의 적용을 제시하였다. 그리고 진화연산으로 얻은 신뢰할 수 있는 결과를 기반으로, primer 정보의 적절한 활용이 우수한 solution 도출에 도움이 된다는 것을 확인했고, 이와는 별도로 ε을 조정하면 적은 계산 비용으로도 대등한 수준의 결과를 얻을 수 있음을 밝혔다. 앞으로는, 입력된 sequence들간을 구별하기 위해 만들어진 probe 뿐만 아니라 생물학적인 연구를 통해 축적되어있는 genome DB와 결과를 비교하고, 현재의 웹사이트를 보완하여 많은 생물학적인 parameter를 받아들일 수 있도록 수정할 예정이다.

감사의 글

이 논문은 과학기술부 국가지정연구실사업(NRL)에 의하여 지원되었음.

참고문헌

[1] S.-Y. Shin, I.-H. Lee, and B.-T. Zhang, "Microarray probe design using ε -multi-objective evolutionary algorithms with thermodynamic criteria," Lecture Notes in Computer Science, vol. 3907, pp. 184-195, 2006, evoWorkshops 2006 (in print).

[2] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multi-objective optimizatin," Evolutionary Computation, vol. 10, no. 3, pp. 263-282, 2002.

[3] Michael D. Gadberry, Simon T. Malcomber, Andrew N. Doust, Elizabeth A. Kellogg, "Primaclade－a flexible tool to find conserved PCR primers across multiple species," Bioinformatics 21(7), pp. 1263-1264, 2005.

[4] Chenna, Ramu, Sugawara, Hideaki, Koike, Tadashi, Lopez, Rodrigo, Gibson, Toby J, Higgins, Desmond G, Thompson, Julie D., "Multiple sequence alignment with the Clustal series of programs," Nucleic Acids Research 31(13), pp. 3497-3500, 2003.

[5] S. Rozen, H. Skaletsky, "Primer3 on the WWW for general users and for biologist programmers," Krawetz S, Misener S (eds) Bioinformatics Methods and Protocols: Methods in Molecular Biology, pp 365-386, 2000.

[6] M. Zuker, "Mfold web server for nucleic acid folding and hybridization prediction," Nucleic Acids Research, vol. 31, no. 13, pp. 3406-3415, 2003.

[7] W. J. Kent, "BLAT－the BLAST-like alignment tool," Genome Research, vol. 12, no. 4, pp. 656-664, 2002.