Published in final edited form as:
IEEE Trans Biomed Eng. 2014 March ; 61(3): 928–937. doi:10.1109/TBME.2013.2292588.

Common Copy Number Variation Detection From Multiple Sequenced Samples

Junbo Duan,
Department of Biomedical Engineering, Xi'an Jiaotong University, Xi'an 710049, China

Hong-Wen Deng, and
Department of Biomedical Engineering and Department of Biostatistics and Bioinformatics, Tulane University, New Orleans, LA 70118 USA

Yu-Ping Wang* **[Senior Member, IEEE]**
Department of Biomedical Engineering and Department of Biostatistics and Bioinformatics, Tulane University, New Orleans, LA 70118 USA

Junbo Duan: junbo.duan@mail.xjtu.edu.cn; Hong-Wen Deng: hdeng2@tulane.edu

Abstract

Common copy number variations (CNVs) [1] are small regions of genomic variations at the same loci across multiple samples, which can be detected with high resolution from next-generation sequencing (NGS) technique. Multiple sequencing data samples are often available from genomic studies; examples include sequences from multiple platforms and sequences from multiple individuals. By integrating complementary information from multiple data samples, detection power can be potentially improved. However, most of current CNV detection methods often process an individual sequence sample, or two samples in an abnormal versus matched normal study; researches on detecting common CNVs across multiple samples have been very limited but are much needed. In this paper, we propose a novel method to detect common CNVs from multiple sequencing samples by exploiting the concurrency of genomic variations in read depth signals derived from multiple NGS data. We use a penalized sparse regression model to fit multiple read depth profiles, based on which common CNV identification is formulated as a change-point detection problem. Finally, we validate the proposed method on both simulation and real data, showing that it can give both higher detection power and better break point estimation over several published CNV detection methods.

Index Terms

Copy number variation (CNV); ℓ_0 norm penalty; model selection; next generation sequencing (NGS); Schur complement; structured sparse modeling; the 1000 genomes project

*wpy@tulane.edu.

I. Introduction

Genetic factors play an important role in the development of a disease. It has been reported that there are tens of thousands of genetic disorders (http://www.ncbi.nlm.nih.gov/Omim/mimstats.html). Among various mutations of human genomes, the copy number variation (CNV) is a sort of structural variation (SV) frequently observed in genomes. CNV is generally referred as a duplication or deletion of DNA sequence of length larger than 1 kbp [2]. Similar duplication and deletion events also occur in somatic cells, which are termed copy number alterations in oncology. There are evidences that CNV can convey human phenotypes from sporadic diseases [3]. It is believed that, when a CNV region harbors a dosage-sensitive segment, the gene expression level varies, and consequently leading to an abnormal phenotype [4].

Fluorescence *in situ* hybridization, and more recently, array comparative genomic hybridization (aCGH) techniques have been widely used to detect CNVs, but with a low resolution of about 5~10 Mbp and 200 bp to 10~25 kbp [5], [6], respectively. In the last few years, next generation sequencing (NGS) technologies allow the screening of human genomes at an unprecedented resolution. NGS platforms produce millions or billions of short reads from shotgun sequencing, and these short reads can be used for *de novo* assembly [7], single nucleotide polymorphism (SNP) calling [8], SV detection [9], etc.

Since the work of Korbel *et al.* [10], Mills *et al.* [11], great efforts have been made to develop CNV detection methods from NGS data [5], [12]–[22]. We recently conducted a comparative study of several prominent CNV detection methods [23], and proposed a robust detection method with a novel sparse regression model [24]. The CNV detection methods from NGS can be mainly divided into four categories [25]: depth of coverage (DOC)- or read depth (RD)-based, paired-end mapping (PEM)-based, split read-based, and assembly-based methods. The canonical procedure of DOC-based method usually consists of the following seven steps: 1) Map (or align) sequencing reads (singled end or paired end) to a reference genome (*e.g.*, NCBI37/hg19) by using short sequencing mapping tools, *e.g.*, Bowtie [26], MAQ [27]. These mapping tools usually output SNP and short indel callings as byproducts. Mapping loci, as well as mapping quality, are stored in a SAM file, or a compacted BAM file. 2) Calculate the so-called RD signal. RD is the read count within a fix-sized nonoverlapping bins [5], [12], or a sliding window [13]. 3) Normalize the RD signal. For example, a GC-content correction is usually performed [28] to reduce bias. 4) Segment normalized RD signal into regions with different depths. Because NGS is characterized by short-gun sequencing, RD reflects copy number status. Most of RD loci are in normal regions; a plateau in RD profile reflects the copy gain status, while a basin reflects the copy loss status. Classical segmentation algorithms, such as the circular binary segmentation (CBS) [29] and hidden Markov model have been employed [14], [20]. 5) Determine copy number status of each segment by a statistical hypothesis testing, *e.g.*, event-wise testing (EWT) [5]. A hypothesis testing assumes that an RD obeys the Poisson [30] or negative-binomial distribution [20]. 6) Merge consecutive segments that share the same copy number status [12]. 7) Call CNVs, including CNV type (gain or loss), starting locus and length, copy number status. We note that not each aforementioned step is necessary. For example, for those methods that detect CNVs from only one sample [5], [21],

Step 3) is necessary, but for those case-control methods [12], [13], i.e., both abnormal and matched normal samples, Step 3) is not needed. CNV-seq and EWT [5] do not use Step 4), i.e., segmentation algorithms.

RD signals are very noisy because of several factors: sequencing error, mapping error (multiple mapping, mismatching), and the presence of SNPs and indels. Therefore, most of the aforementioned methods, which focus on detection from single sample, or two samples as in aCGH platforms, achieve low power and high false positive rate. A simple approach to improve detection power is to increase the sequencing coverage. However, this approach will be at greater sequencing cost; an alternative cost-efficient study design is to sequence large samples with a medium or low coverage [8]. With the continuous decrease of sequencing cost, sequencing multiple times or with multiple platforms will be increasingly used. Multiple sequencing can reduce the system error introduced by an individual sample or platform, with the potential to improve the detection power [31]. In addition, it was reported that different complex diseases might share the same common CNVs [32], which could be detected with multiple sequencing data. To this end, the detection of common CNVs from multiple sequencing data is much needed, promising to give higher detection power.

In this paper, we propose a novel method to detect CNVs from multiple sequencing data. The proposed method first fits multiple RD profiles from several samples using an $\ell 0$ penalized least-square regression model. Then, CNV detection is obtained with a statistical testing. The objective function used in the regression model consists of two terms. The first is a data fitting term, i.e., the least squares of fitting error, and the second is a penalty term, i.e., the $\ell 0$ norm of change-points of the RD signal. Different from our previous work [33], the new objective function explores the concurrency of common CNVs by using a block-wise $\ell 0$ norm. So, if a CNV at the same locus shows up across multiple sequencing data, a block-wise sparse vector (regressor) will be obtained in the regression model.

In order to validate and evaluate the performance of our model, we tested it on simulated data and compared with cn.MOPS [30], a method recently proposed to detect CNVs from multiple samples. We also tested our method on real data, and compared it with two other methods: CNVnator [21] and EWT [5]. The real data include a 17 replication raw sequencing dataset of a HapMap sample, a mapped dataset from multiple sequencing platforms, and a mapped dataset of a family trio from a pilot study of the 1000 Genomes Project.

II. Methods

A. Modeling

After preprocessing, we have an array of signals $\boldsymbol{y}_j = [y_{1j}, y_{2j}, \ldots, y_{Mj}]^T$, $j = 1, \ldots, N$, i.e., the RD profile of the jth sample, where M is the number of bins. In the following, we model the CNV detection as a change-point detection problem, similar to our previous work based on aCGH platform [34]. “From statistical point of view, a change-point is defined as a point (either an index or a spatial location) before which a random sequence follows a distribution with certain parameter(s), and after which the random sequence follows another distribution

[or the same distribution as before but with different parameter(s)]" [34]. Specifically, for our problem, the copy number status can be reflected by the RD signal, where the regions with different copy numbers follow different distributions (especially the means). Therefore, we define a change-point as the locus where the mean of RD signal changes significantly. In other words, a change-point corresponds to the boundary of a CNV region.

To detect change-points from these RD signals, we use the linear combination of a set of step functions (***A***) to approximate $\boldsymbol{y}_j$ (see Fig. 1)

$$\boldsymbol{y}_j \approx \boldsymbol{A}\boldsymbol{x}_j \quad (1)$$

where ***A*** is an Heaviside dictionary, consisting of a set of step functions defined at different loci

$$\boldsymbol{A}=\begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (2)$$

and $\boldsymbol{x}_j$ reflects the amplitudes of change-points. Because there is a limited number of change-points (boundary of CNVs) in the genome, $\boldsymbol{x}_j$ is assumed to be a sparse vector, i.e., most entries of $\boldsymbol{x}_j$ are zeros. The ℓ-0 norm, $\|\boldsymbol{x}_j\|_0$, has been widely used to measure the sparsity of a vector. Therefore, the detection of CNVs from $\boldsymbol{y}_j$ can be modeled as [35]

$$\boldsymbol{x}_j(\lambda)=\arg\min_{\boldsymbol{x}_j}\|\boldsymbol{y}_j-\boldsymbol{A}\boldsymbol{x}_j\|^2+\lambda\|\boldsymbol{x}_j\|_0 \quad (3)$$

where the penalty parameter λ controls the tradeoff between the data fitting and the regularization term. The regularization or penalty term incorporates prior knowledge about data into the model [24]. It is obvious that large λ yields less number of change-points, and vice-versa.

The sparse model (1) for finding CNVs from individual sample can be generalized to multiple samples $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N]$ as follows:

$$\boldsymbol{X}(\lambda)=\arg\min_{\boldsymbol{X}}\|\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{X}\|^2+\lambda\|\boldsymbol{X}\|_0 \quad (4)$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N]$ contains the coefficients of all N samples, and its element x_{ij} corresponds to the amplitude of the change-point at ith locus in jth sample. $\|\boldsymbol{X}\|_0$ is the row-wise λ-0 norm of matrix ***X*** (see Fig. 1), which is defined as

$$\|\boldsymbol{X}\|_0=\sum_{i=1}^{M}\iota(\tilde{\boldsymbol{x}}_i) \quad (5)$$

$$\iota(\tilde{\boldsymbol{x}}_i)=\begin{cases} 0, & \tilde{\boldsymbol{x}}_i=\mathbf{0} \\ 1, & \text{otherwise.} \end{cases} \quad (6)$$

The entries of $\boldsymbol{x}_i\tilde{}$ reflect the amplitudes of change-points. If all samples have a change-point at the locus i, $\boldsymbol{x}_i\tilde{}$ should be a nonzero vector. Therefore, $\iota(\boldsymbol{x}_i)\tilde{} = 1$ indicates that there is a change-point at ith locus, while $\iota(\boldsymbol{x}_i)\tilde{} = 0$ indicates no change-points. By introducing the row-wise ℓ0 norm, we force the change-points from multiple samples to be aligned at the same locus.

B. Optimization Algorithm

The problem in (3) can be solved approximately with greedy algorithms such as matching pursuit [36], orthogonal matching pursuit (OMP) [37], and orthogonal least squares (OLS) [38]. However, when matrix $\boldsymbol{A}$ is highly correlated, OMP- and OLS-based methods might fail. Therefore, we propose the single best replacement (SBR) algorithm that we developed [39] to solve the problem in (3). To solve the more general problem in (4), in this paper, we extend the algorithm and develop a continuation block-wise SBR (CBSBR). The acceleration issue is discussed in Appendix A.

1) Estimating a Solution When λ is Fixed—First, we introduce an active set $\mathcal{A} \subseteq \{1, 2, \ldots, M\}$, which indicates the activity status of the columns of matrix $\boldsymbol{A}$. The presence of an index i in the active set indicates that the ith column of $\boldsymbol{A}$ is used to fit the data $\boldsymbol{Y}$, and the ith row of $\boldsymbol{X}$ is set to be nonzeros (see Fig. 1). If the active set $\mathcal{A}$ is known, we can have the extracted submatrix $\boldsymbol{A}_{\mathcal{A}} = \{\boldsymbol{a}_i \,|\, i \in \mathcal{A}\}$, and the least-square solution

$$\boldsymbol{X}_{\mathcal{A}}=\boldsymbol{A}_{\mathcal{A}}^{\dagger}\boldsymbol{Y} \quad (7)$$

where $\boldsymbol{A}_{\mathcal{A}}^{\dagger}=(\boldsymbol{A}_{\mathcal{A}}^{T}\boldsymbol{A}_{\mathcal{A}})^{-1}\boldsymbol{A}_{\mathcal{A}}^{T}$ is the Moore–Penrose inverse [40] of $\boldsymbol{A}_{\mathcal{A}}$. $\boldsymbol{X}$ can be reconstructed from $\boldsymbol{X}_{\mathcal{A}}$ by inserting zeros into nonactive rows. Because the least-square solution $\boldsymbol{X}_{\mathcal{A}}$ normally has no zero row, $\|\boldsymbol{X}\|_0$ is equal to the cardinality of $\mathcal{A}$, i.e., the number of elements in $\mathcal{A}$, which is denoted by k. In summary, given $\mathcal{A}$ with k elements, we consecutively have $\boldsymbol{A}_{\mathcal{A}}$ of size $M \times k$, $\boldsymbol{X}_{\mathcal{A}}$ of size $k \times N$, and the equivalent cost function

$$\mathcal{J}_{\mathcal{A}}(\lambda) \triangleq \|\boldsymbol{Y}-\boldsymbol{A}_{\mathcal{A}}\boldsymbol{X}_{\mathcal{A}}\|^2+\lambda\|\boldsymbol{X}_{\mathcal{A}}\|_0=\mathcal{E}_{\mathcal{A}}+\lambda k. \quad (8)$$

where

$$\mathcal{E}_{\mathcal{A}} \triangleq \|\boldsymbol{Y}-\boldsymbol{A}_{\mathcal{A}}\boldsymbol{X}_{\mathcal{A}}\|^2. \quad (9)$$

To have a better fitting quality, more columns are needed to be active, suggesting larger k. Therefore, the basic idea of the proposed algorithm is to include/exclude an index into/from $\mathcal{A}$ iteratively, which is called a single replacement

$$\mathscr{A} \bullet i = \begin{cases} \mathscr{A} \cup \{i\}, & \text{if } i \notin \mathscr{A} \\ \mathscr{A} \backslash \{i\}, & \text{otherwise.} \end{cases} \quad (10)$$

We initialize the block-wise SBR (BSBR) algorithm with $\mathcal{A} = \emptyset$. Then, at each iteration, we test all possible *M* single replacements, and find the "best" update in a deepest decent manner, i.e., the replacement that can most reduce cost function $\mathcal{J}$. The iteration terminates when there is no single replacement that can further reduce the cost function. Since the number of possible single replacement in BSBR algorithm is finite, it terminates after a finite number of iterations, and converge to a local minimum [39]. Table I summarizes the BSBR algorithm, and the acceleration is discussed in Appendix A. Since the total number of single replacement increases linearly with respect to *M*, the computation complexity is proportional to *M*. For the sample size *N*, Equations (26) and (30) in Appendix A show that *N* has very limited impact on the computational complexity.

2) Model Selection—Since the penalty parameter λ controls the tradeoff between the data-fitting and penalty term in the problem of (4), it is of great importance to have a good estimate of λ. When the prior knowledge of a system is known, λ could be estimated from system parameters [39]. When the prior knowledge is unknown, classical model selection methods can be used. Since different models correspond to different sets of active columns, direct estimation of λ could be replaced by the selection of candidate models. Therefore, we have to estimate a set of candidate models first.

A simple method to estimate a set of candidate models is to solve the problem (4) at a uniform or logarithm grid of λ. However, this method is not efficient, since the solution is a set of piecewise constant [41] with respect to λ. When the grid is too fine, we may find the same candidate model at several different λ's; when the grid is too coarse, we may miss some candidate models. We adopt the idea that we proposed in [41] to find the critical values of λ. Then, with the calls of BSBR algorithm at these critical λ's, all candidate models can be obtained.

We note that $\boldsymbol{X}(+\infty) = 0$ and $\boldsymbol{X}(0) = \boldsymbol{A}^{+}\boldsymbol{Y}$. Inspired by the homotopy algorithm [42]–[44], the proposed CBSBR algorithm starts with $\lambda = +\infty$, and then decreases λ adaptively. At each critical λ, BSBR is called to estimate current solution, and next λ is calculated according to current solution. Specifically, if at current λ_q active set is $\mathcal{A}_q$, then the next maximal $\lambda_{q+1} < \lambda_q$ at which the active set changes [41] should satisfy

$$\mathscr{J}_{\mathscr{A}_q}(\lambda_q) = \mathscr{J}_{\mathscr{A}_q \cup l}(\lambda_{q+1}). \quad (11)$$

Up to a few manipulations of (8) and (11), the value of λ_{q+1} is

$$\lambda_{q+1} = \max_{i \notin \mathscr{A}_q} \{\mathscr{E}_{\mathscr{A}_q} - \mathscr{E}_{\mathscr{A}_q \cup l}\}. \quad (12)$$

Table II summarizes the procedure of the CBSBR algorithm. The stopping condition could be either the least-square error $\varepsilon_{\mathcal{A}}$ or sparsity level *k* reaches some predefined value.

Once all possible models are known, we could use classical model selection methods such as cross-validation [45], *L*-curve [46], Akaike information criterion [47], and Schwarz information criterion (SIC) [48]. Since the SIC has been proven to deliver robust estimate as shown in our previous work [34], [49], we use the SIC in our experiments.

For a candidate model $\boldsymbol{X}(\lambda_q)$, its SIC is defined as

$$\mathrm{SIC}(q)=k\ln(M)+\frac{\|\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{X}(\lambda_q)\|^2}{\sigma_n^2} \quad (13)$$

where k is the row-wise ℓ0 norm of $\boldsymbol{X}(\lambda_q)$ as introduced in Section II-A, and σ_n^2 is the variation of noise in $\boldsymbol{Y}$, which can be estimated from non-CNV regions. The best candidate model can be obtained with the lowest SIC, giving the optimal tradeoff between fitting quality and model complexity k.

After the best model is selected, CNVs can be detected from the recovered RD signal $\boldsymbol{AX}(\lambda_{q^*})$ by thresholding with upper and lower cutoff values, where q^* is the index of best model. These cutoff values can affect sensitivity and specificity, so we employ the histogram (or distribution) of the RD signal to estimate them. Since sequencing may have different coverages, we estimate the two cutoff values (upper and lower) for each RD signal, such that the portion of tail areas is lower than a predefined value.

We briefly summarize the proposed method. First, we use the CBSBR algorithm, which calls BSBR algorithm iteratively, to estimate a set of solutions with respect to continuous change of penalty parameter λ. Second, we use the SIC method to select the best model from this set of solutions. Finally, the regions of the smoothed RD signal that are above upper threshold or below lower threshold are considered to be duplications or deletions of CNV regions, respectively.

III. Results

In order to test the performance of the proposed method, we compared it with three published methods, *i.e.*, CNVnator [21], EWT [5], and cn.MOPS [30]. The first two are dedicated to detect CNVs from single data sample, while the last one focuses on the detection from multiple data.

We tested the performance of the proposed method on both simulated and real data from human subject studies. In simulations, we tested in terms of sensitivity, specificity, and break point locus estimation, with respect to different dispersion level, sample size, and the CNV frequency across samples. In real-data processing, we tested three real datasets. The first dataset consists of raw data of a HapMap TSI human subject, whose genome was sequenced 17 times repeatedly; the second dataset consists of mapped data of a HapMap YRI human subject with three popular sequencing platforms; and the third dataset consists of mapped data of a family trio in the 1000 Genomes Project pilot study.

All computations were carried out on a PC with a dual-core 2.8-GHz x86 64-bits processor and 8-GB memory, and the computational time to screen a whole genome with the resolution of 1 kbp is approximately 10 h with a peak memory usage of 700 MB.

A. Simulations

First, we tested the robustness of the proposed method with respect to different dispersion level. Because of the presence of SNPs, indels, base pair calling error, alignment error, as well as the randomness of shotgun sequencing, RD signals are often poorly modeled to be the Poisson distribution, where the mean and variance are assumed to be equal. Instead, the negative binomial distribution was proposed [20], [50], where a positive dispersion parameter d is introduced to tune the variance to mean ratio (see Appendix B). When d is close to 0, the negative binomial distribution approaches to the Poisson distribution asymptotically; when d increases, the variation to mean ratio increases accordingly, indicating higher dispersion level. Fig. 2 presents an example when the dispersion level d is 0.1. The RD signals are displayed with blue dots, and the smoothed signals are shown as red curves.

The simulated RD signals consist of N signals generated with bin size of 100 bp, and with average RD of 200. Since the CNVs of large sizes are relatively easy to detect, we simulated nine artificial CNVs of small sizes ranging from 1 to 4 kbp, with copy numbers 0, 1, 3~6 (see Fig. 2). Note that the copy number 1 (heterozygous deletion) and 3 (heterozygous duplication) are especially challenging to detect, since they are close to normal copy number 2 [17].

A number of dispersion parameter $d = 0.001, 0.01, 0.1, 0.5, 1$ with sample size $N = 6$ were tested. At each dispersion level, 1000 random's $\boldsymbol{Y}$ were simulated to follow the negative binomial distribution, and the mean was fixed to 200. Then, the proposed method was employed to process each data. The detected CNVs were compared with the ground truth, yielding the receiver operating characteristic (ROC) as shown in the left panel of Fig. 3. An ROC shows the tradeoff between sensitivity and specificity, where each point represents the average of 1000 replications under a given dispersion level. A point in the ROC shows the true positive rate (TPR, equivalent to sensitivity or recall) versus false positive rate (FPR, equivalent to 1-specificity). TPR and FPR are defined in unit of base pair (bp). The TPR is the ratio between number of base pairs in the detected CNVs that overlap with the ground truth, and those in the ground truth; The FPR is the ratio between number of base pairs in the detected CNVs that do not overlap with the ground truth, and those not in the ground truth.

It is shown in the left penal of Fig. 3 that with an increase in dispersion level, the detection quality degenerates, especially when the dispersion level is greater than 0.1. Further, studies show that, when the dispersion level is greater than 0.1, nearby CNVs are difficult to be distinguished, which are considered to merge together. As a result, FPR increases significantly.

To further evaluate the performance of detection, we tested the precision of detecting break points. For two overlapping CNVs (i.e., one is from the detection output, while the other is from the ground truth), the differences between the left and right loci are calculated as the

break point estimation error. The mean and standard deviation are calculated after 1000 replications. As shown in the right panel of Fig. 3, the accuracy of break point estimation improves with the decrease of the dispersion level. As mentioned previously, when the dispersion level is greater than 0.1, nearby CNVs merge together; so the quality of break point estimation degenerates greatly. For this reason, the right panel of Fig. 3 only displays the estimation error with the dispersion level of 0.1, 0.01, and 0.001.

To test the effect of sample size N on the FPR, TPR and break point estimation, datasets consisting of 1, 2, 4, 6, and 8 samples were simulated, with the dispersion level being fixed to 0.1. Both the ROC and error bar for the estimations of break points are plotted in Fig. 4. The results indicate that multiple samples can improve detection power, decrease false detection rate, and improve break point localization. To keep the detection power above 0.9, in other simulations, the sample size N is fixed to 6.

The cn.MOPS method was recently developed to discover CNVs from multiple samples. However, it could only detect variations when the copy numbers at the same loci are different across samples. So in the previous simulations, where the copy number status across multiple samples is the same, cn.MOPS detected no variations. In order to compare the performance of the proposed method with that of cn.MOPS, a Bernoulli parameter p is introduced to denote the CNV frequency across multiple samples. Therefore, the previous simulations correspond to the case where $p = 1$. In the following simulations, data with $p = 0.2$, 0.5, and 0.8 were generated. Fig. 2 presents an example when the Bernoulli parameter p is 0.8. Fig. 5 presents the ROC plots, which summarizes the results of this comparative study. For the proposed method, when the dispersion level is relatively high ($d > 0.1$), the detection power decreases significantly. But when $d \leq 0.1$, the influence of p on the detection power is very limited. When $d(\leq 0.1)$ is fixed, $p = 0.8$, 0.5, 0.2 yields the lowest, second lowest, and highest FPR, respectively. For cn.MOPS, $p = 0.5$ yields higher power when compared with $p = 0.2$ and 0.8. Note that at each fixed configuration (p, d), the proposed method always yields higher TPR than that of cn.MOPS, while at the cost of increased FPR.

B. Real Data Processing

1) Multiple Sequencing Data From a Single Individual—The genome of a HapMap TSI sample (NA20755) was sequenced 17 times by a Solexa Genome Analyzer II with low coverage ranging from 0.13 to 0.21, and with read length 37 bp [51]. We downloaded this dataset from the FTP of DNA Data Bank of Japan at http://www.ddbj.nig.ac.jp/, and then mapped pair-end reads to the NCBI36/hg18 reference genome with Bowtie [26], allowing no more than two mismatches with “best” option on. RD signals were extracted from the 17 alignment BAM files using the tool in cn.MOPS package, yielding 23 data matrices, corresponding to 22 autosomes and X chromosome. To have an average of read counts around 100, the bin size was set to 25 kbp. The RD bias caused by GC-content was corrected by using the method introduced by Abyzov *et al.* [21], with the GC-content profile of RDXplorer [5]. The final step of normalization is to correct biases caused by the coverage difference, so each RD signal was scaled such that the mean of read counts is 100.

Since the 17 sequencing data are from the same sample, at a given bin normalized RD signals should have no significant difference, that is, copy numbers should be equal. So, we designed the following analysis procedure. Each autosome was processed with three replications. In each replication, we randomly implanted three CNVs of size 75, 150, and 200 kbp into the chromosome. For non-CNV regions, RD signals from the autosome were used; while for CNV regions, RD signals from X chromosome were used according to the copy number. A CNV could be either a gain (copy number 3) or a loss (copy number 1) with equal probability. For a gain (or loss) CNV, the copy number could be either 3 (or 1) or 2 with a frequency of 0.9 and 0.1, respectively.

Data matrices were processed by both cn.MOPS and the proposed method. For cn.MOPS, the upper and lower cutoff values were set to 0.5 and −0.5 respectively; the minimum number of segments, a CNV should span was set to 3. The CBS [29] was used as the segmentation algorithm; and other parameters were set as default. For the proposed method, five percent of tail area was used to determine the cutoff values, the minimal size of a segment was set to 3 bins, and the noise variance σ_n^2 in (13) was estimated along whole genome, since the portion of CNV regions is negligible with respect to whole genome.

The results are listed in Table III, where the TPR and FPR for cn.MOPS are 0.49 and 8e-4, while for the proposed method they are 0.68 and 5e-4, respectively. We also repeated experiments, with the Bernoulli parameter $p = 0.5$ and 0.2 (the probability of copy number 3 or 1, versus $1 - p$ the probability of copy number 2). As shown in this table, for the proposed method, the TPR/FPR increases/decreases with the increase of p. For cn.MOPS, it performs well with $p = 0.5$, where the variation of copy numbers across samples reaches the peak value.

2) Sequencing Data From Multiple Platforms—The 1000 Genomes Project ([51], http://www.1000genomes.org/) provides data acquired from multiple sequencing platforms. We downloaded the mapped sequencing data (BAM files) of a YRI sample with Corelli/HapMap ID NA19240, with the platforms of Roche 454, Illumina SLX, and ABI SOLiD. The short reads were mapped already by using SSAHA2 [52], MAQ [27], and Corona lite [53], respectively.

The RD signals were extracted from BAM files by using the SAMtools [27]. Bin size was set to 1 kbp, yielding mean RD 13 for 454, 84 for SLX, and 220 for SOLiD, respectively. Since the coverage is significantly different, each RD signal after the GC-content correction was scaled such that the mean value is 100.

CNVnator [21] and EWT [5] were used to process the three data one by one. The bin size of CNVnator is set to 300 bp, and the consecutive window number of EWT is set to 8. The detected CNVs are displayed in the first two panels in Fig. 6. Since data from the three platforms are complementary, the union of the three platforms was calculated for both CNVnator and EWT. The intersection of two unions (i.e., CNVnator and EWT union) is displayed in the right panel of Fig. 6 with the label “CNVnatorEWT.”

The overlaps of CNVs from the three methods (CNVna-torEWT, cn.MOPS, and the proposed one) are displayed in the right panel of Fig. 6. The overlap is measured in the unit of 100 bp. If two units share more than one base pair overlap, it counts for one overlapped unit. It is shown that the detection power of cn.MOPS and the proposed method is 73% and 88%, respectively. The detection powers of CNVnator with 454, SLX, and SOLiD platforms are 72%, 73%, and 64%; and those of EWT are 58%, 65%, and 85%, respectively. The result demonstrates that the detection power is increased with multiple platforms over single one.

3) Sequencing Data From Multiple Individuals—We downloaded the aligned sequencing data of chromosome 21 of a CEU family trio from the 1000 Genomes Project pilot study. This family trio has European ethnicity, and consists of NA12891 the father, NA12892 the mother, and NA12878 the daughter.

We used the same procedure with the same settings as in the previous study to analyze data. Fig. 7 shows the comparison of CNVs detected by the proposed method, EWT, and CNVnator, respectively. In the first row, three Venn diagrams are displayed with respect to three samples. It is shown that, on average 70% of detected CNVs with the proposed method overlap with at least one alternative method (CNVnator or EWT). For CNVnator and EWT, this value is 52% and 43%, respectively, suggesting that the proposed method can detect CNVs with higher consistency. In the second row, three diagrams are displayed with respect to different methods. On average, 82% CNVs detected by the proposed method show up at least twice among the three CEU samples; this value is 65% for CNVnator, and 32% for EWT. These results suggest that the proposed method can detect more common CNVs.

Fig. 8 shows the comparison of CNVs detected by the proposed method and cn.MOPS. The “proposed(1)/(2)/(3)” represents CNVs detected with the proposed method that show up at least once/twice/three times among three samples, respectively. It is shown that 82% CNVs detected by the proposed method show up at least twice among three samples, and 63% among all three. It is also shown that 36% overlaps with the result of cn.MOPS.

Further study shows that, among the CNVs that show up twice in the three family members, 11% are shared by the father and mother, 22% by the father and daughter, and 67% by the mother and daughter. Interestingly, the Venn diagrams in Fig. 7 labeled with “CNVnator” and “proposed” show that the daughter (NA12878) shares more CNVs with her mother (NA12892) than her father (NA12891), suggesting that the daughter is ge-nomically closer to her mother than her father. This is consistent with the results of Magi *et al.* [54], and ours reported before [55].

IV. Conclusion and Discussion

In this paper, we proposed a method to detect common CNVs from multiple NGS data, which are often measured with different experimental replications, multiple platforms, and multiple individuals. By introducing the row-wise ℓ_0 norm in the regression model, the concurrency of CNV across multiple samples can be captured. We also proposed a novel numerical method, i.e., CBSBR algorithm to solve the regression model and further used the

model to fit multiple RD signals, based on which CNVs can be inferred. We note that CBSBR can be used to other applications where matrix $\boldsymbol{A}$ might have other structure than the Heaviside dictionary used in this paper.

We tested the performance of the proposed method on both simulated and real data. The simulation results show the performance of detection in terms of the dispersion level and the CNV frequency across samples. It suggests that the detection power, false positive rate, and break point loci estimation can all be improved with the increase of sample size. For real-data processing, we analyzed the 17 replication data from a single individual and multiple datasets from three popular sequencing platforms. The results indicate that by integrating multiple datasets with complementary information, the proposed method outperforms single-sample-based methods. We also used the data from a family trio, suggesting that the proposed method can be used to discover genetic ethnicity in terms of CNVs.

From both simulation and real data analysis, we found that CNVs with a high frequency across samples can be detected with the proposed method, which falls into our expectation. Since the proposed method is based on the concurrency of CNVs across multiple samples, higher the frequency that a CNV shows up, easier will it be detected. On the contrary, CNVs with a moderate frequency of presence can be detected by cn.MOPS by capturing the variation of copy number across samples. However, if copy numbers are the same across samples, these CNVs can be missed by cn.MOPs. So, when data are highly redundant (i.e., replication data), we recommend the use of the proposed method. When data are from different samples, we recommend that these two methods should be combined to take full advantage of both approaches.

Both DOC and PEM signature can provide useful and complementary information [5], [54]: the DOC-based methods can detect large-size events, while the PEM-based methods can detect small-size events with better precision of finding break point locus. Therefore, some approaches were proposed to combine both methods. For example, CNVer [15] and cnvHiTSeq [50] combined both DOC and PEM signatures to improve break point detection. He *et al.* [56] used discordant read pairs and unmapped reads that span break points to detect CNVs, and the precision of detecting CNV break point can reach as high as base pair level. So our future work will consider the incorporation of multiple signatures into the model, which could further improve the CNV detection accuracy.

The MATLAB code of the proposed CBSBR algorithm can be downloaded at http://www.mathworks.com/matlabcentral/fileexchange/36518-continuation-block-wise-sparse-approximation. Other codes are available upon request.

Acknowledgments

This work was supported in part by the National Institutes of Health and National Science Foundation Grant.

References

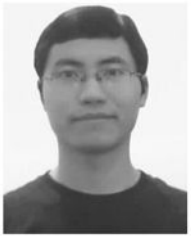
1. Shlien A, Malkin D. Copy number variations and cancer. Genome Med. 2009; 1(6):1–9. [PubMed: 19348688]

2. Freeman JL, Perry GH, Feuk L, Redon R, McCarroll SA, Altshuler DM, Aburatani H, Jones KW, Tyler-Smith C, Hurles ME, Carter NP, Scherer SW, Lee C. Copy number variation: new insights in genome diversity. Genome Res. Aug.2006 16:949–961. [PubMed: 16809666]
3. Inoue K, Lupski JR. Molecular mechanisms for genomic disorders. Annu Rev Genom Hum Genet. 2002; 3:199–242.
4. Stankiewicz P, Lupski JR. Structural variation in the human genome and its role in disease. Annu Rev Med. 2010; 61:437–455. [PubMed: 20059347]
5. Yoon S, Xuan Z, Makarov V, Ye K, Sebat J. Sensitive and accurate detection of copy number variants using read depth of coverage. Genome Res. Sep.2009 19:1586–1592. [PubMed: 19657104]
6. Urban AE, Korbel JO, Selzer R, Richmond T, Hacker A, Popescu GV, Cubells JF, Green R, Emanuel BS, Gerstein MB, Weissman SM, Snyder M. High-resolution mapping of DNA copy alterations in human chromosome 22 using high-density tiling oligonucleotide arrays. Proc Nat Acad Sci USA. Mar; 2006 103(12):4534–4539. [PubMed: 16537408]
7. Lin Y, Li J, Shen H, Zhang L, Papasian CJ, Deng H-W. Comparative studies of *de novo* assembly tools for next-generation sequencing technologies. Bioinformatics. Aug; 2011 27(15):2031–2037. [PubMed: 21636596]
8. Nielsen R, Paul JS, Albrechtsen A, Song YS. Genotype and SNP calling from next-generation sequencing data. Nat Rev Genet. Jun; 2011 12(6):443–451. [PubMed: 21587300]
9. Medvedev P, Stanciu M, Brudno M. Computational methods for discovering structural variation with next-generation sequencing. Nat Methods. Nov.2009 6:S13–S20. [PubMed: 19844226]
10. Korbel JO, Urban AE, Affourtit JP, Godwin B, Grubert F, Simons JF, Kim PM, Palejev D, Carriero NJ, Du L, Taillon BE, Chen Z, Tanzer A, Saunders ACE, Chi J, Yang F, Carter NP, Hurles ME, Weissman SM, Harkins TT, Gerstein MB, Egholm M, Snyder M. Paired-end mapping reveals extensive structural variation in the human genome. Science. Oct.2007 318:420–426. [PubMed: 17901297]
11. Mills RE, Walter K, Stewart C, Handsaker RE, Chen K, Alkan C, Abyzov A, Yoon SC, Ye K, Cheetham RK, Chinwalla A, Conrad DF, Fu Y, Grubert F, Hajirasouliha I, Hormozdiari F, Iakoucheva LM, Iqbal Z, Kang S, Kidd JM, Konkel MK, Korn J, Khurana E, Kural D, Lam HYK, Leng J, Li R, Li Y, Lin C-Y, Luo R, Mu XJ, Nemesh J, Peckham HE, Rausch T, Scally A, Shi X, Stromberg MP, Stutz AM, Urban AE, Walker JA, Wu J, Zhang Y, Zhang ZD, Batzer MA, Ding L, Marth GT, McVean G, Sebat J, Snyder M, Wang J, Ye K, Eichler EE, Gerstein MB, Hurles ME, Lee C, McCarroll SA, Korbel JO. 1000 Genomes Project. Mapping copy number variation by population-scale genome sequencing. Nature. Feb; 2011 470(7332):59–65. [PubMed: 21293372]
12. Chiang DY, Getz G, Jaffe DB, O'Kelly MJT, Zhao X, Carter SL, Russ C, Nusbaum C, Meyerson M, Lander ES. High-resolution mapping of copy-number alterations with massively parallel sequencing. Nat Methods. Jan.2009 6:99–103. [PubMed: 19043412]
13. Xie C, Tammi MT. CNV-seq, a new method to detect copy number variation using high-throughput sequencing. BMC Bioinformatics. 2009; 10:1–9. [PubMed: 19118496]
14. Simpson JT, McIntyre RE, Adams DJ, Durbin R. Copy number variant detection in inbred strains from short read sequence data. Bioinformatics. Feb; 2010 26(4):565–567. [PubMed: 20022973]
15. Medvedev P, Fiume M, Dzamba M, Smith T, Brudno M. Detecting copy number variation with mated short reads. Genome Res. Nov; 2010 20(11):1613–1622. [PubMed: 20805290]
16. Waszak SM, Hasin Y, Zichner T, Olender T, Keydar I, Khen M, Stutz AM, Schlattl A, Lancet D, Korbel JO. Systematic inference of copy-number genotypes from personal genome sequencing data reveals extensive olfactory receptor gene content diversity. PLoS Comput Biol. 2010; 6:1–20.
17. Kim T-M, Luquette LJ, Xi R, Park PJ. rSW-seq: algorithm for detection of copy number alterations in deep sequencing data. BMC Bioinformatics. 2010; 11:1–13. [PubMed: 20043860]
18. Ivakhno S, Royce T, Cox AJ, Evers DJ, Cheetham RK, Tavaré S. CNAseg–a novel framework for identification of copy number changes in cancer from second-generation sequencing data. Bioinformatics. Dec; 2010 26(24):3051–3058. [PubMed: 20966003]
19. Boeva V, Popova T, Bleakley K, Chiche P, Cappo J, Schleiermacher G, Janoueix-Lerosey I, Delattre O, Barillot E. Control-free calling of copy number alterations in deep-sequencing data using GC-content normalization. Bioinformatics. Jan; 2011 27(2):268–269. [PubMed: 21081509]

20. Miller CA, Hampton O, Coarfa C, Milosavljevic A. ReadDepth: a parallel R package for detecting copy number alterations from short sequencing reads. PLoS ONE. 2011; 6:1–7.
21. Abyzov A, Urban AE, Snyder M, Gerstein M. CNVnator: An approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. Genome Res. Jun; 2011 21(6):974–984. [PubMed: 21324876]
22. Gusnanto A, Wood HM, Pawitan Y, Rabbitts P, Berri S. Correcting for cancer genome size and tumour cell content enables better estimation of copy number alterations from next-generation sequence data. Bioinformatics. Jan; 2012 28(1):40–47. [PubMed: 22039209]
23. Duan J, Zhang J-G, Deng H-W, Wang Y-P. Comparative studies of copy number variation detection methods for next generation sequencing technologies. Plos One. 2013; 8(3):1–12.
24. Duan J, Zhang J-G, Deng H-W, Wang Y-P. CNV-TV: A robust method to discover copy number variation from short sequencing reads. BMC Bioinformatics. 2013; 14(150):1–12. [PubMed: 23323762]
25. Teo SM, Pawitan Y, Ku CS, Chia KS, Salim A. Statistical challenges associated with detecting copy number variations with next-generation sequencing. Bioinformatics. 2012; 28(21):2711–2718. [PubMed: 22942022]
26. Langmead B, Trapnell C, Pop M, Salzberg SL. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol. 2009; 10(3):1–10.
27. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R. 1000 Genome Project Data Processing Subgroup. The sequence alignment/map format and SAMtools. Bioinformatics. Aug; 2009 25(16):2078–2079. [PubMed: 19505943]
28. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR, Boutell JM, Bryan J. Accurate whole human genome sequencing using reversible terminator chemistry. Nature. Nov.2008 456:53–59. [PubMed: 18987734]
29. Olshen AB, Venkatraman ES, Lucito R, Wigler M. Circular binary segmentation for the analysis of array-based DNA copy number data. Biostatistics. Oct.2004 5:557–72. [PubMed: 15475419]
30. Klambauer G, Schwarzbauer K, Mayr A, Clevert D-A, Mitterecker A, Bodenhofer U, Hochreiter S. cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. Nucleic Acids Res. Feb.2012 :1–14.
31. Ratan A, Miller W, Guillory J, Stinson J, Seshagiri S, Schuster SC. Comparison of sequencing platforms for single nucleotide variant calls in a human sample. PLoS ONE. 2013; 8(2):1–10.
32. Moreno-De Luca D, Consortium SGENE, Mulle JG, Kaminsky EB, Sanders SJ, Gene S-TAR, Myers SM, Adam MP, Pakula AT, Eisenhauer NJ, et al. Deletion 17q12 is a recurrent copy number variant that confers high risk of autism and schizophrenia. Amer J Human Genet. 2010; 87(5):618–630. [PubMed: 21055719]
33. Duan, J.; Zhang, J-G.; Lefante, J.; Deng, H-W.; Wang, Y-P. Simons Simplex Collection Genetics Consortium. Detection of copy number variation from next generation sequencing data with total variation penalized least square optimization. Proc. IEEE Int. Conf. Bioinformatics Biomed. Workshop; Atlanta, GA, USA. 2011. p. 3-12.
34. Chen J, Wang Y-P. A statistical change point model approach for the detection of DNA copy number variations in array CGH data. IEEE/ACM Trans Comput Biol Bioinformatics. Oct; 2009 6(4):529–541.
35. Duan, J.; Soussen, C.; Brie, D.; Idier, J. Détection conjointe de discontinuites d'ordres différents dans un signal par minimisation de critére L2-L0. presented at the Actes 22e coll; Sep. 2009; Dijon, France: GRETSI;
36. Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries. IEEE Trans Signal Process. Dec; 1993 41(12):3397–3415.
37. Pati, Y.; Rezaiifar, R.; Krishnaprasad, P. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. Proc. 27th Asilomar Conf. Signals, Syst. Comput; 1993. p. 40-44.
38. Chen S, Billings SA, Luo W. Orthogonal least squares methods and their application to non-linear system identification. Int J Control. 1989; 50(5):1873–1896.
39. Soussen C, Idier J, Brie D, Duan J. From Bernoulli-Gaussian deconvolution to sparse signal restoration. IEEE Trans Signal Process. Oct; 2011 59(10):4572–4584.

40. Bernstein, D. Matrix Mathematics. Princeton, NJ, USA: Princeton Univ. Press; 2009.
41. Duan, J.; Soussen, C.; Brie, D.; Idier, J. A continuation approach to estimate a solution path of mixed L2-L0 minimization problems. Proc. Signal Process. Adapt. Sparse Struct. Represent; Saint-Malo, France. Apr. 2009; p. 1-6.
42. Osborne MR, Presnell B, Turlach BA. A new approach to variable selection in least squares problems. IMA J Numer Anal. 2000; 20(3):389–403.
43. Efron B, Hastie T, Johnstone I, Tibshirani R. Least angle regression. Ann Statist. 2004; 32(2):407–499.
44. Malioutov, DM.; Cetin, M.; Willsky, AS. Homotopy continuation for sparse signal representation. Proc. IEEE Int. Conf. Acoust., Speech, Signal Process; Philadephia, PA, USA. Mar. 2005; p. 733-736.
45. Golub GH, Heath M, Wahba G. Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics. May; 1979 21(2):215–223.
46. Hansen P. Analysis of discrete ill-posed problems by means of the L-curve. SIAM Rev. 1992; 34:561–580.
47. Akaike H. A new look at the statistical model identification. IEEE Trans Autom Control. Dec; 1974 AC-19(6):716–723.
48. Schwarz G. Estimating the Dimension of a Model. Ann Statist. 1978; 6:461–464.
49. Markon KE, Krueger RF. An empirical comparison of information-theoretic selection criteria for multivariate behavior genetic models. Behav Genet. 2004; 34(6):593–610. [PubMed: 15520516]
50. Bellos E, Johnson MR, Coin LJM. cnvHiTSeq: integrative models for high-resolution copy number variation detection and genotyping using population sequencing data. Genome Biol. 2012; 13(12):1–11.
51. The 1000 Genomes Project Consortium. A map of human genome variation from population-scale sequencing. Nature. Oct; 2010 467(7319):1061–1073. [PubMed: 20981092]
52. Ning Z, Cox AJ, Mullikin JC. SSAHA: A fast search method for large DNA databases. Genome Res. Oct; 2001 11(10):1725–1729. [PubMed: 11591649]
53. Applied biosystems, [Online]. Available: http://www.umassmed.edu/uploadedFiles/nemo/Landing_Pages/07_S3_July_Corona_Lite_intro_final.pdf
54. Magi A, Benelli M, Yoon S, Roviello F, Torricelli F. Detecting common copy number variants in high-throughput sequencing data by using JointSLM algorithm. Nucl Acids Res. May; 2011 39(10):1–9. [PubMed: 20805246]
55. Duan, J.; Zhang, J-G.; Deng, H-W.; Wang, Y-P. Detection of common copy number variation with application to population clustering from next generation sequencing data. Proc. IEEE Int. Conf. Eng. Med. Biol. Soc; San Diego, CA, USA. 2012. p. 1246-1249.
56. He D, Furlotte N, Eskin E. Detection and reconstruction of tandemly organized de novo copy number variations. BMC Bioinformatics. 2010; 11(Suppl 11):1–8. [PubMed: 20043860]
57. Goussard, Y.; Demoment, G.; Idier, J. A new algorithm for iterative deconvolution of sparse spike trains. Proc. IEEE Int. Conf. Acoust., Speech, Signal Process; Albuquerque, NM, USA. Apr. 1990; p. 1547-1550.

Biographies



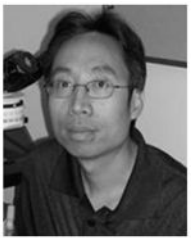
Junbo Duan received the B.S. degree in information engineering, and the M.S. degree in communication and information system from Xi'an Jiaotong University, Xi'an, China, in 2004 and 2007, respectively, and the Ph.D. degree in signal processing from the Université Henry Poincaré, Nancy, France, in 2010.

After his graduation, he was a Postdoctoral Fellow in the Department of Biomedical Engineering, and the Department of Biostatistics and Bioinformatics, Tulane University, New Orleans, LA, USA, until 2013. He is currently an Assistant Professor in the Department of Biomedical Engineering, Xi'an Jiaotong University. His major research interests are in probabilistic approaches to inverse problems in biomedical engineering and bioinformatics.

Hong-Wen Deng received the Bachelor's degree in ecology and environmental biology and the Master's degree in ecology and entomology from Peking University, Beijing, China. He received the Master's degree in mathematical statistics and the Ph.D. degree in quantitative genetics from the University of Oregon, Eugene, OR, USA.

He was a Postdoctoral Fellow in the Human Genetics Center, University of Texas in Houston, where he conducted postdoctoral research in molecular and statistical population/quantitative genetics. He also served as a Hughes Fellow in the Institute of Molecular Biology at the University of Oregon. He previously served as a Professor of medicine and biomedical sciences at Creighton University Medical Center, Professor of orthopedic surgery and basic medical science and the Franklin D. Dickson/Missouri Endowed Chair in Orthopedic Surgery at the School of Medicine of University of Missouri-Kansas City. He is currently the Chair of Tulane Department of Biostatistics and Bioinformatics and the Director of Center of Bioinformatics and Genomics, Tulane University, New Orleans, LA, USA. He has widely published more than 400 peer-reviewed articles, ten book chapters, and three books. His area of interest is in the genetics of osteoporosis and obesity.

Dr. Deng received multiple National Institutes of Health R01 awards and multiple honors for his research.



Yu-Ping Wang (SM'06) received the B.S. degree in applied mathematics from Tianjin University, Tianjin, China, in 1990, and the M.S. degree in computational mathematics and the Ph.D. degree in communications and electronic systems from Xi'an Jiaotong University, Xi'an, China, in 1993 and 1996, respectively.

After his graduation, he had visiting positions at the National University of Singapore, Singapore, and the Washington University Medical School, St. Louis, MO, USA. From 2000 to 2003, he was a Senior Research Engineer at Perceptive Scientific Instruments, Inc., and then at Advanced Digital Imaging Research, LLC, Houston, TX, USA. In the fall of 2003, he returned to academia as an Assistant Professor of computer science and electrical engineering at the University of Missouri-Kansas City. He is currently an Associate Professor in the Department of Biomedical Engineering and the Department of Biostatistics and Bioinformatics, Tulane University, New Orleans, LA, USA, and a member of Tulane Center of Bioinformatics and Genomics and Tulane Cancer Center. His research interests include interdisciplinary biomedical imaging and bioinformatics, where he has more than 120 peer reviewed publications.

Dr. Wang has served on numerous program committees and National Science Foundation/ National Institutes of Health review panels. He is an Associate Editor for several journals including *Journal of Neuroscience Methods*, and was a member of Machine Learning for Signal Processing technical committee of the IEEE Signal Processing Society.

Appendix A. Acceleration of the CBSBR Algorithm

At each iteration of CBSBR, we have to calculate the least-square solution (7) repeatedly. Since the computational complexity of the inverse of matrix $A_{\mathcal{A}}$ is $\mathcal{O}(k^3)$, where k is the column dimension of $A_{\mathcal{A}}$, computational burden is heavy when considering large number of single replacements. In this Appendix, we present an acceleration algorithm, which is based on the iterative solution of nested least-square problems [57].

The main idea is to calculate the increment of the cost function $\mathcal{J}_{\mathcal{A}\cup l} - \mathcal{J}_{\mathcal{A}}$ without explicit calculation of the inverse of $A_{\mathcal{A}\cup l}$ in inclusion cases, or $\mathcal{J}_{\mathcal{A}\setminus l} - \mathcal{J}_{\mathcal{A}}$ without the inversion of $A_{\mathcal{A}\setminus l}$ in exclusion cases.

To be specific, given $\mathcal{A}$ and $l \notin \mathcal{A}$, let's define

$$\begin{gathered}\boldsymbol{G}=[\boldsymbol{A}_{\mathscr{A}},\boldsymbol{a}_l]\\ \boldsymbol{\phi}_{\mathscr{A}}=(\boldsymbol{A}_{\mathscr{A}}^T\boldsymbol{A}_{\mathscr{A}})^{-1}\\ \boldsymbol{\phi}_{\boldsymbol{G}}=(\boldsymbol{G}^T\boldsymbol{G})^{-1}\\ \boldsymbol{w}=\boldsymbol{A}^T\boldsymbol{a}_l.\end{gathered}$$

From (7) and (9), up to a few manipulations, we have

$$\begin{gathered}\mathscr{J}_{\mathscr{A}}=\mathscr{E}_{\mathscr{A}}+\lambda k\\ =\|\boldsymbol{Y}\|^2-\sum_{i=1}^{N}\boldsymbol{y}_i^T\boldsymbol{A}_{\mathscr{A}}\boldsymbol{\phi}_{\mathscr{A}}\boldsymbol{A}_{\mathscr{A}}^T\boldsymbol{y}_i+\lambda k\end{gathered}\quad(14)$$

$$\begin{gathered}\mathscr{J}_{\boldsymbol{G}}=\mathscr{J}_{\mathscr{A}\cup l}\\ =\mathscr{E}_{\boldsymbol{G}}+\lambda(k+1)\end{gathered}\quad(15)$$

$$=\|\boldsymbol{Y}\|^2-\sum_{i=1}^{N}\boldsymbol{y}_i^T\boldsymbol{G}\boldsymbol{\phi}_{\boldsymbol{G}}\boldsymbol{G}^T\boldsymbol{y}_i+\lambda(k+1).\quad(16)$$

From Schur complement lemma [40], we have

$$\boldsymbol{\phi}_{\boldsymbol{G}}=\begin{bmatrix}\boldsymbol{\phi}_{11} & \boldsymbol{\phi}_{12}\\ \boldsymbol{\phi}_{21} & \phi_{22}\end{bmatrix}\quad(17)$$

with

$$\boldsymbol{\phi}_{11}=\boldsymbol{\phi}_{\mathscr{A}}+\phi_{22}^{-1}\boldsymbol{\phi}_{12}\boldsymbol{\phi}_{21}\quad(18)$$

$$\boldsymbol{\phi}_{12}=-\phi_{22}\boldsymbol{\phi}_{\mathscr{A}}\boldsymbol{w}\quad(19)$$

$$\boldsymbol{\phi}_{21}=\boldsymbol{\phi}_{12}^T\quad(20)$$

$$\phi_{22}=(\boldsymbol{a}_l^T\boldsymbol{a}_l-\boldsymbol{w}^T\boldsymbol{\phi}_{\mathscr{A}}\boldsymbol{w})^{-1}.\quad(21)$$

Furthermore,

$$\boldsymbol{\phi}_{\boldsymbol{G}}=\begin{bmatrix}\boldsymbol{\phi}_{\mathscr{A}} & \boldsymbol{0}\\ \boldsymbol{0}^T & 0\end{bmatrix}+\phi_{22}\begin{bmatrix}\boldsymbol{\phi}_{\mathscr{A}}\boldsymbol{w}\\ -1\end{bmatrix}\begin{bmatrix}\boldsymbol{w}^T\phi_{\mathscr{A}}^T & -1\end{bmatrix}.\quad(22)$$

1. For inclusion cases, calculate $\mathscr{J}_{\mathscr{A}\cup l}-\mathscr{J}_{\mathscr{A}}$ when $\boldsymbol{\phi}_{\mathscr{A}}$ is known: From (14), (16), and (22)

$$\mathcal{J}_{\mathcal{A}\cup l}-\mathcal{J}_{\mathcal{A}}=\sum_{i=1}^{N}\boldsymbol{y}_i^T\left(\boldsymbol{A}_{\mathcal{A}}\boldsymbol{\phi}_{\mathcal{A}}\boldsymbol{A}_{\mathcal{A}}^T-\boldsymbol{G}\boldsymbol{\phi}_{\boldsymbol{G}}\boldsymbol{G}^T\right)\boldsymbol{y}_i+\lambda \quad (23)$$

$$=\sum_{i=1}^{N}\boldsymbol{y}_i^T\left(-\phi_{22}\boldsymbol{G}\begin{bmatrix}\boldsymbol{\phi}_{A}\boldsymbol{w}\\-1\end{bmatrix}\cdot\begin{bmatrix}\boldsymbol{w}^T\boldsymbol{\phi}_{\mathcal{A}}^T & -1\end{bmatrix}\boldsymbol{G}^T\right)\boldsymbol{y}_i+\lambda \quad (24)$$

$$=-\phi_{22}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{G}\begin{bmatrix}\boldsymbol{\phi}_{\mathcal{A}}\boldsymbol{w}\\-1\end{bmatrix}\right)^2+\lambda \quad (25)$$

$$=-\phi_{22}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{A}_{\mathcal{A}}\boldsymbol{\phi}_{\mathcal{A}}\boldsymbol{w}-\boldsymbol{y}_i^T\boldsymbol{a}_l\right)^2+\lambda. \quad (26)$$

2. For exclusion cases, calculate $\mathcal{J}_{\mathcal{A}\setminus l}-\mathcal{J}_{\mathcal{A}}$ when $\boldsymbol{\phi}_{\mathcal{A}}$ is known

From (18), we have

$$\boldsymbol{\phi}_{\mathcal{A}}=\boldsymbol{\phi}_{11}-\phi_{22}^{-1}\boldsymbol{\phi}_{12}\boldsymbol{\phi}_{21}. \quad (27)$$

By substituting $\mathcal{A}'$ and l' with $\mathcal{A}\cup l$ and l, respectively, from (25) and (19), we have

$$\mathcal{J}_{\mathcal{A}'\setminus l'}-\mathcal{J}_{\mathcal{A}'}=\mathcal{J}_{\mathcal{A}}-\mathcal{J}_{\boldsymbol{G}} \quad (28)$$

$$=\phi_{22}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{G}\begin{bmatrix}\boldsymbol{\phi}_{\mathcal{A}}\boldsymbol{w}\\-1\end{bmatrix}\right)^2-\lambda \quad (29)$$

$$\begin{aligned}&=\phi_{22}^{-1}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{G}\begin{bmatrix}\phi_{22}\boldsymbol{\phi}_{\mathcal{A}}\boldsymbol{w}\\-\phi_{22}\end{bmatrix}\right)^2-\lambda\\&=\phi_{22}^{-1}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{G}\begin{bmatrix}-\boldsymbol{\phi}_{12}\\-\phi_{22}\end{bmatrix}\right)^2-\lambda\\&=\phi_{22}^{-1}\sum_{i=1}^{N}\left(\boldsymbol{y}_i^T\boldsymbol{G}\begin{bmatrix}\boldsymbol{\phi}_{12}\\\phi_{22}\end{bmatrix}\right)^2-\lambda.\end{aligned} \quad (30)$$

Finally, the acceleration version of the BSBR algorithm can be summarized as follows: 1) test all replacements with the help of (26) for inclusion cases and (30) for exclusion cases, then find the best, or deepest-descent one. 2) Update φ with the help of (22) for inclusion cases or (27) for exclusion cases.

Appendix B. Negative Binomial Distribution and Dispersion Parameter d

The negative binomial distribution describes the probability distribution of the number of successes *S* in a sequence of Bernoulli trials before *f* number of failures occurs. If *b* denotes the probability of success in each trial, we say the number of successes *S* follows the negative binomial distribution with parameter *f* and *b*, with probability mass function

$$p_{\mathrm{NB}}(s;f,b)=Pr(S=s)=\begin{pmatrix} s+f-1 \\ s \end{pmatrix}(1-b)^f b^s. \quad (31)$$

The mean and variance of *S* are $\mu_{\mathrm{NB}}=\frac{bf}{1-b}$ and $\sigma^2_{\mathrm{NB}}=\frac{bf}{(1-b)^2}$, respectively. For fixed μ_{NB} and σ^2_{NB}, we have estimate $f=\frac{\mu^2_{\mathrm{NB}}}{\sigma^2_{\mathrm{NB}}-\mu_{\mathrm{NB}}}$. The dispersion parameter *d* is defined as the reciprocal of *f*, i.e., $d=\frac{1}{f}=\frac{\sigma^2_{\mathrm{NB}}-\mu_{\mathrm{NB}}}{\mu^2_{\mathrm{NB}}}$. So for fixed μ_{NB}, the variance to mean ratio $\frac{\sigma^2_{\mathrm{NB}}}{\mu_{\mathrm{NB}}}=d\mu_{\mathrm{NB}}+1$ could be tuned by dispersion parameter *d*.

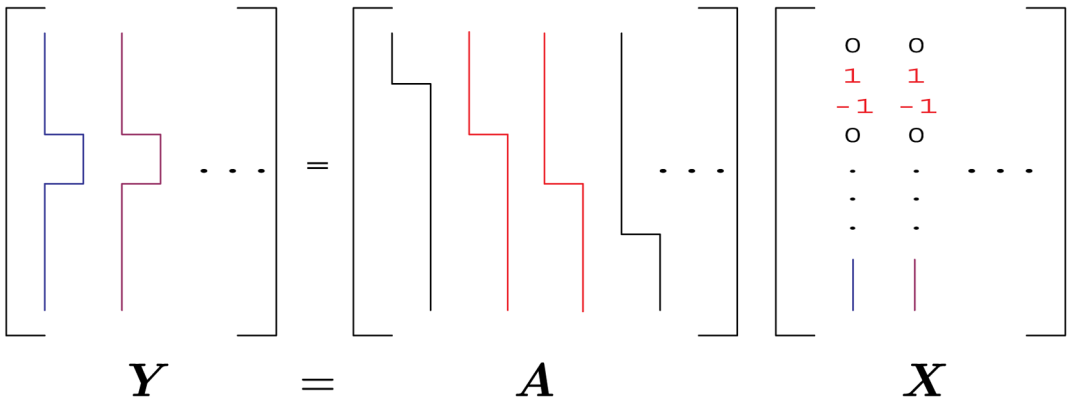


Fig. 1.

Schematic demonstration of the approximation of the RD signals ***Y*** with a sparse vector ***X*** and a set of step functions. The columns of ***Y*** and ***A*** (step functions) are displayed in wave forms. Red color indicates active status. The row-wise ℓ-0 norm of matrix ***X*** is 2. In this case, there are two nonzero rows in ***X*** corresponding to the loci of two change-points in ***Y***.

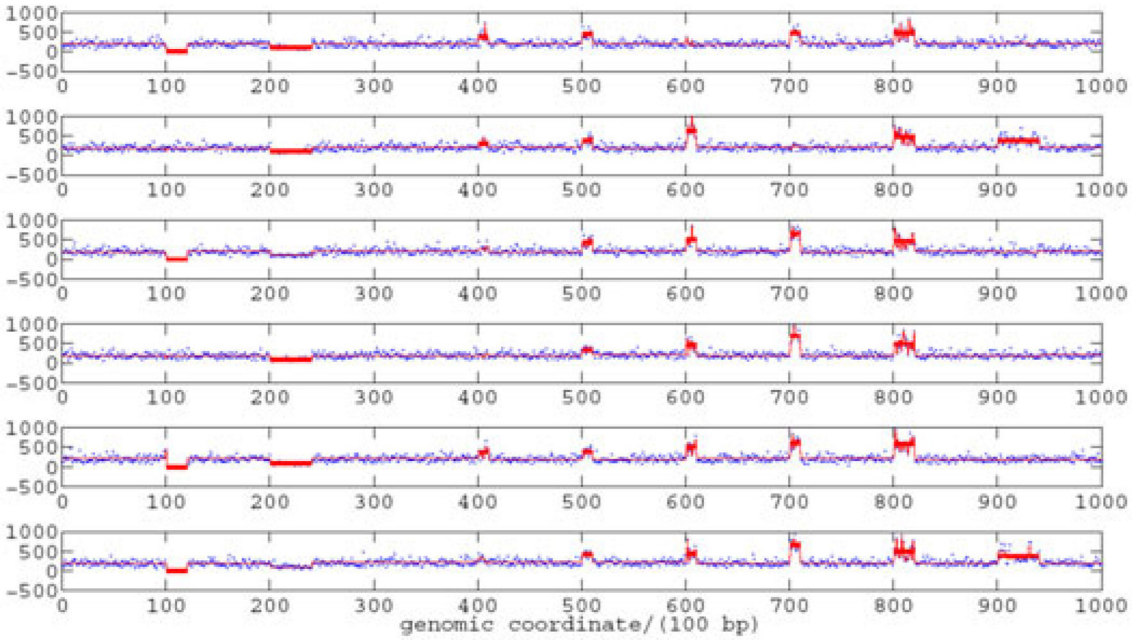


Fig. 2.

Example of CNV detection by the proposed method. Blue dots are RD signals with dispersion level $d = 0.1$ and Bernoulli parameter $p = 0.8$, red thin lines are smoothed signals, and red thick lines are detected CNV regions.

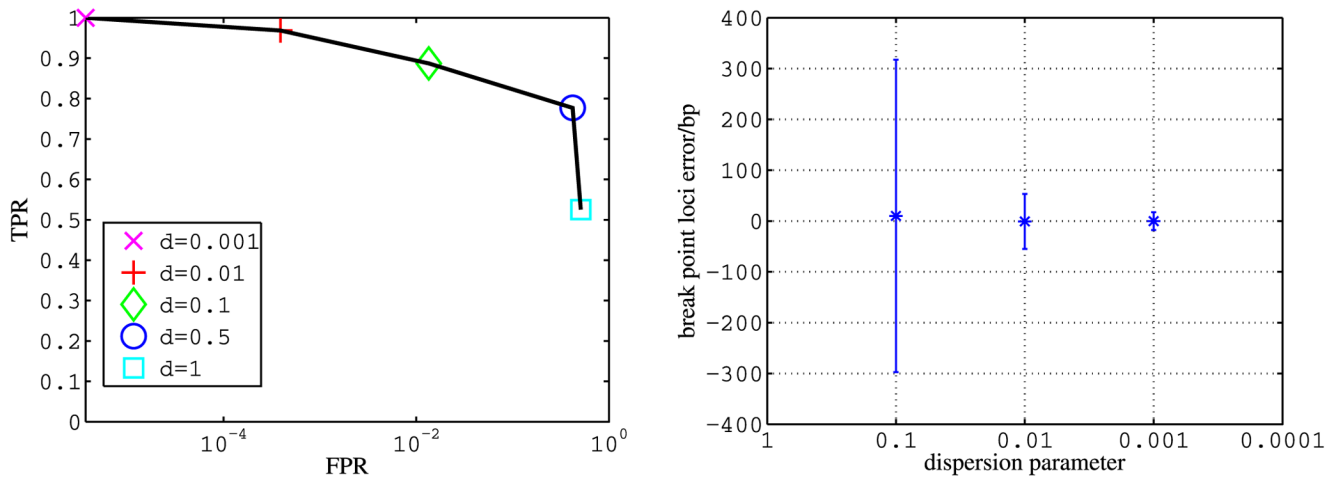


Fig. 3.

ROC (left) and error bar (right) plot with different dispersion level. It is shown that with the decrease of dispersion level, FPR decreases, while TPR increases, and break point loci estimation improves.

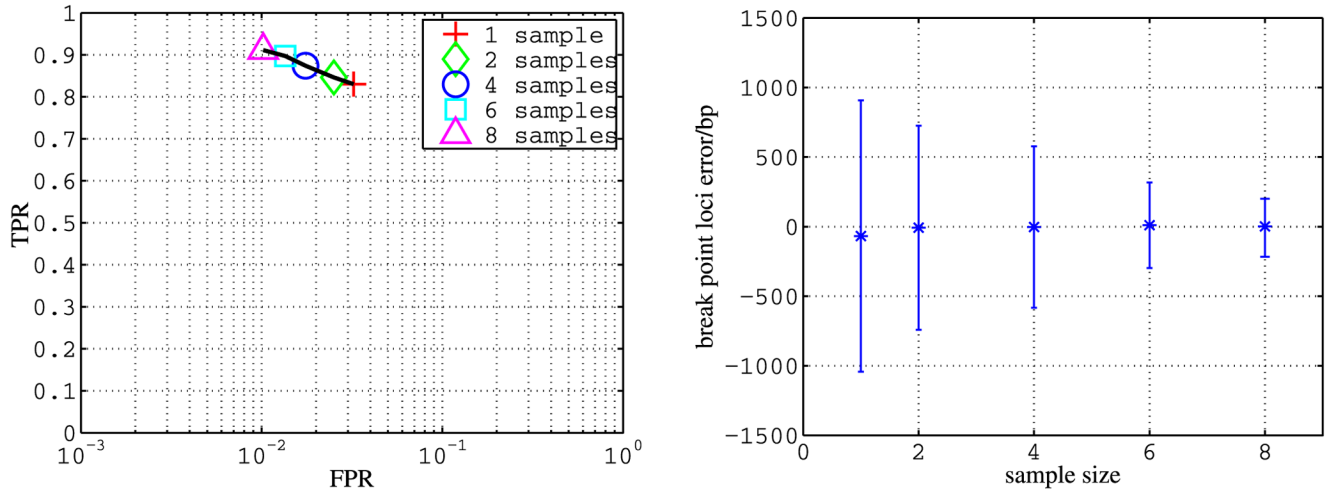


Fig. 4.

ROC (left) and error bar (right) plot with different sample size. It is shown that with the increase of sample size, FPR decreases, while TPR increases, and break point loci estimation improves.

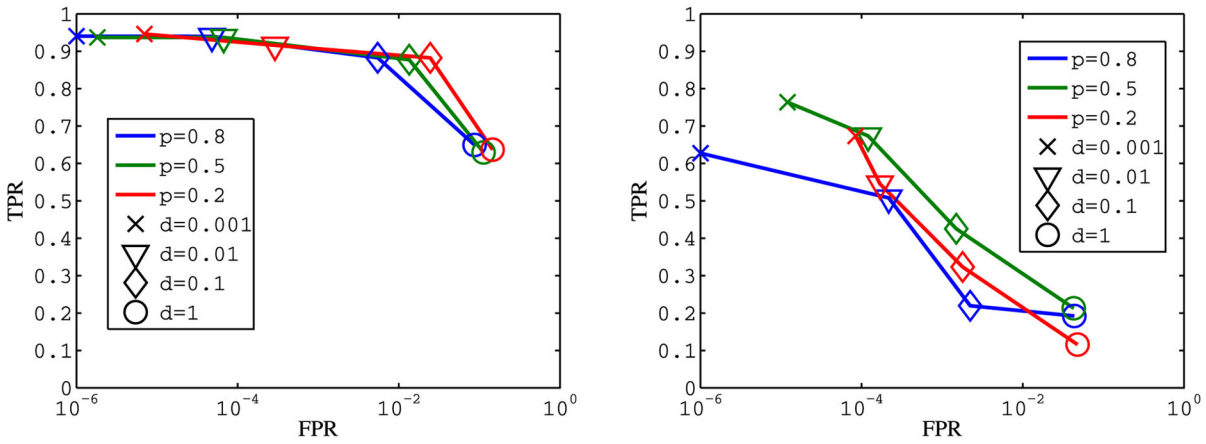


Fig. 5.

ROC plots with different Bernoulli parameter *p* and dispersion level *d* with the proposed method (left) and cn.MOPS (right). It is shown that the proposed method achieves higher detection power compared with cn.MOPS.

Fig. 6.

Venn diagrams of three platforms. The first two panels show the overlapping of CNVs (in the unit of 100 bp) detected by CNVnator (left) and EWT (middle). In the last panel, the set labeled with “CNVnatorEWT” is the intersection of CNVs detected with CNVnator and EWT.

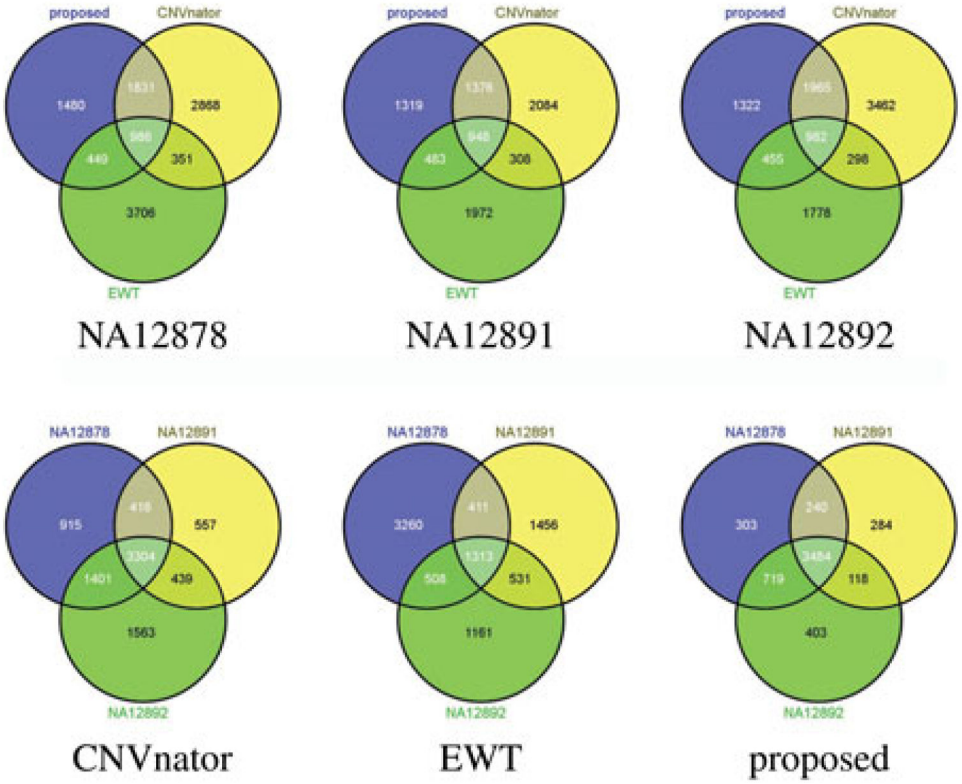


Fig. 7.

Venn diagrams of the CEU family trio. The top three depict the overlaps of CNVs detected from three CEU samples; the lower three depict the results from CNVnator, EWT, and the proposed method.

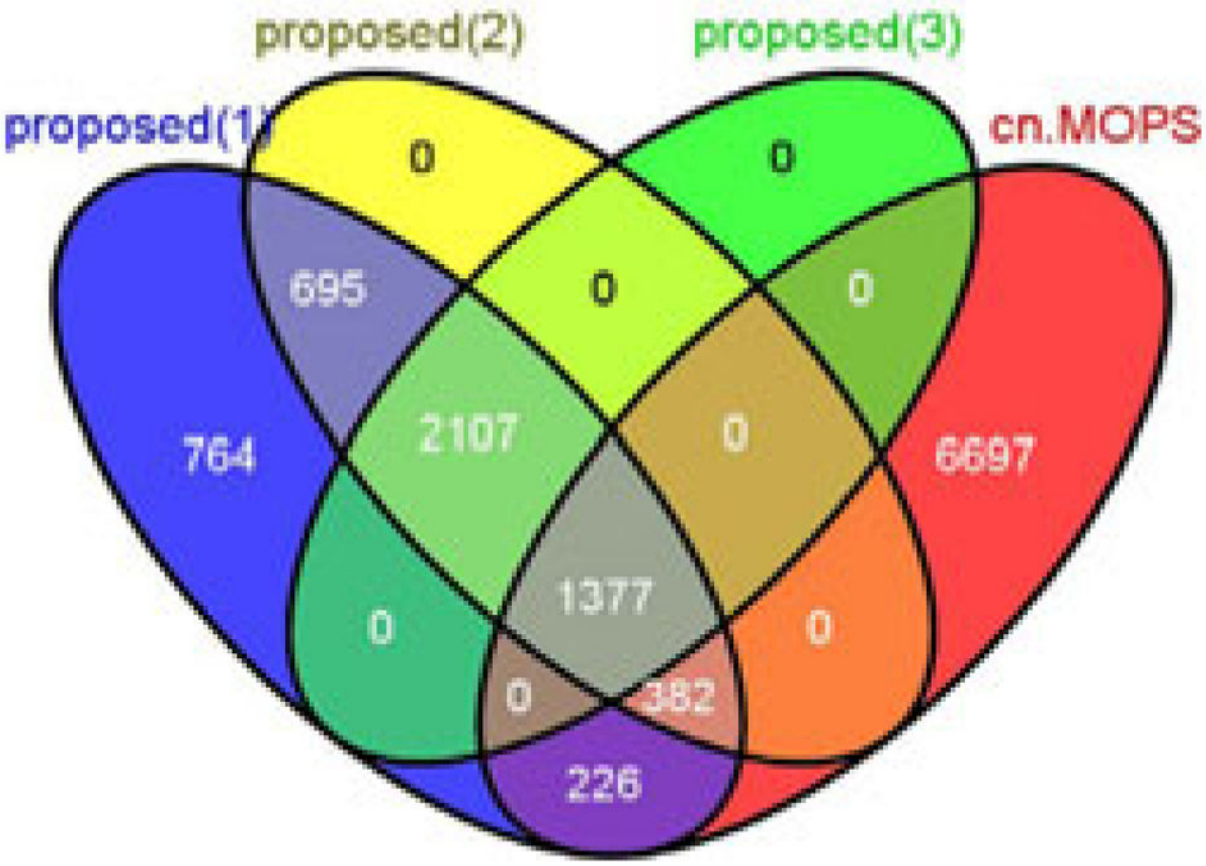


Fig. 8.

Comparison with cn.MOPS. “Proposed(1)/(2)/(3)” represents the CNVs detected by the proposed method that show up at least once/twice/three times among the three CEU samples.

TABLE I

BSBR Algorithm

Input:	$\boldsymbol{A}$, $\boldsymbol{Y}$, λ and $\mathcal{A}$
Step1:	Initialization: $k = 0$, calculate $\mathcal{J}_{\mathcal{A}}$ according to (8).
Step2:	For iteration $k > 0$, test all single replacement, find the best $l_k = \arg\min_i \mathcal{J}_{\mathcal{A}\bullet i}$, update $\mathcal{A}$ and $\mathcal{J}_{\mathcal{A}}$.
Step3:	Iterate Step2 until cost function $\mathcal{J}$ no longer decreases.
Step4:	Calculate $\boldsymbol{X}_{\mathcal{A}}$ and $\mathcal{E}_{\mathcal{A}}$ according to Eq. (7) and (9).
Output:	$\boldsymbol{X}$, $\mathcal{E}_{\mathcal{A}}$.

TABLE II

CBSBR Algorithm

Input:	$\boldsymbol{A}$, $\boldsymbol{Y}$.
Step1:	Initialization $\lambda_i = +\infty$, $\mathcal{A}_0 = \varnothing$
Step2:	For iteration $q > 0$, $[\boldsymbol{X}(\lambda_q), \mathcal{E}_{\mathcal{A}_q}] = \mathrm{BSBR}(\boldsymbol{A}, \boldsymbol{Y}, \lambda_q, \mathcal{A}_{q-1})$, update $\mathcal{A}_q$ according to $\boldsymbol{X}(\lambda_q)$, calculate λ_{q+1} according to Eq. (12).
Step3:	Iterate Step2 until stopping condition.
Output:	$\{\mathbf{X}(\lambda_q)\}$, $\{\lambda_q\}$.

Table III

TPR and FPR Performance of the Sequencing Data of NA20755

	Cn.MOPS		Proposed	
	TPR	**FPR**	**TPR**	**FPR**
p=0.9	0.49	8e-4	0.68	5e-4
p=0.5	0.62	35e-3	0.61	1.2e-3
p=0.2	0.50	3.6e-3	0.59	3.3e-3