2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) | 979-8-3503-2447-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/EMBC40787.2023.10340070

# Decentralized Parallel Independent Component Analysis for Multimodal, Multisite Data

Chan Aek Panichvatana, Jiayu Chen, Bradley Baker, Bishal Thapaliya, *Member, IEEE*, Vince Calhoun, *Fellow, IEEE* , Jingyu Liu, *Member, IEEE*

***Abstract*— Large amounts of neuroimaging and omics data have been generated for studies of mental health. Collaborations among research groups that share data have shown increased power for new discoveries of brain abnormalities, genetic mutations, and associations among genetics, neuroimaging and behavior. However, sharing raw data can be challenging for various reasons. A federated data analysis allowing for collaboration without exposing the raw dataset of each site becomes ideal. Following this strategy, a decentralized parallel independent component analysis (dpICA) is proposed in this study which is an extension of the state-of-art Parallel ICA (pICA). pICA is an effective method to analyze two data modalities simultaneously by jointly extracting independent components of each modality and maximizing connections between modalities. We evaluated the dpICA algorithm using neuroimage and genetic data from patients with schizophrenia and health controls, and compared its performances under various conditions with the centralized pICA. The results showed dpICA is robust to sample distribution across sites as long as numbers of samples in each site are sufficient. It can produce the same imaging and genetic components and the same connections between those components as the centralized pICA. Thus our study supports dpICA is an accurate and effective decentralized algorithm to extract connections from two data modalities.**

## I. INTRODUCTION

Large amounts of neuroimaging and omics data have been shared for research on psychological well-being [1]. Neuroimaging is a brain imaging technique widely used to study human brain structure and function, while genomics is analyzed to understand how mutations in genetic sequence affect pathology of human diseases. One commonly studied variation is single nucleotide polymorphisms (SNP). To understand mental illness, data from neuroimaging, multi-omics, cognitive function, behavioral assessment, and other modalities are often collected and tested. Deficits in cognitive function can be a sign of various neurological or psychological disorders [2] along with abnormal behavior associated with various mental disorders. Finding these multidimensional connections is crucial for mental disorder research.

One of the algorithms to analyze interactions between multiple modalities is parallel independent component analysis (pICA) [2]. pICA is an effective method to analyze two data modalities simultaneously by jointly extracting independent components of each modality and maximizing connections between modalities. Compared to other multimodal algorithms, such as canonical correlation analysis, partial least squares, and reduced rank regression [3], pICA allows researchers to select a specific number of components for each modality and balance the independence and connections to prevent overfitting. The current version of pICA, however, only works on centralized data where all the data are located at one site.

It is well-established that shared data analysis by collaboration can improve the power of statistical tests, allowing for larger and more diverse populations than centralized studies can offer [4], [5]. This is also the case in Neuroimaging due to the cost of collecting scans related to particular questions. In Neuroimaging and other fields with complex, privacy-sensitive data, it is often not feasible for a collaborative research study to pool raw data from individual participating sites into a centralized location. Thus, decentralized pICA (dpICA) is proposed here to accommodate such a situation by allowing research sites to perform dpICA without transferring full samples between sites, thus preserving patient privacy while still increasing the population of data available.

One application of pICA from Liu et al. [2] is to identify interactions between brain function and genetic information by analyzing functional magnetic resonance imaging (fMRI) and SNPs and find the related pairs of fMRI/SNP component. Similarly Chen et al. [6] applied the pICA method on structural MRI and SNPs and identified the gray matter volume (GMV) in patients with schizophrenia is reduced in the temporal and parietal junction area [7] where 39 SNPs are associated with. In this paper, we replicated the study of Chen et al. [6] using the proposed dpICA method, and evaluated the validity and reliability of the dpICA method under various setting.

## II. THEORY AND ALGORITHM

### A. Independent Component Analysis - ICA

The goal of ICA is to identify independent component matrices (S) embedded in the data from one modality using a linear decomposition, where each component is maximally independent. Equation (1) shows the general formula of ICA. X is the data $\in \mathbb{R}^{N\times d}$ where $N$ is the number of subjects, and $d$ is the size of the variable. $S$ is the independent component $\in \mathbb{R}^{r\times d}$ where $r$ is the number of components. $A$ is the loading parameter matrix $\in \mathbb{R}^{N\times r}$ . $W$ is the unmixing matrix $\in \mathbb{R}^{r\times N}$. While there are many algorithms to implement ICA, we used Infomax [8] as shown in (2) where $fy(Y)$ is the probability density function of $Y$. $E$ is the expected value. $H$ is the entropy function. Maximization of entropy $H$ is used to maximize the independence among the components contained in $S$.

$$X = A \cdot S; \quad S = W \cdot X; \quad A \quad = W^{-1} \tag{1}$$

$$\max\{H(Y)\} = -E\left[\ln f_y(Y)\right];$$
$$Y = \frac{1}{1+e^{-U}}, U = W \cdot X + W_0 \tag{2}$$

### B. Parallel ICA

The purpose of pICA is to identify independent components of two modalities and additionally to enhance the relation of the two modalities. When applying pICA, principal component analysis (PCA) is first performed to reduce data dimension. As in Equation (3), $X$ is the data $\in \mathbb{R}^{N\times d_x}$ where $N$ is the number of subjects, and $d_x$ is the number of variables. $\Sigma_x^{-1} \cdot V_x^T$ is the projection matrix with whitening in the dimension of number of component ($r_x$)-by-number of subject ($N$). We denote this matrix as Whitening matrix for simplicity. $V_x \cdot \Sigma_x$ is the inverse projection matrix with de-whitening, noted as DeWhitening matrix. Similarly, Equation (4) is PCA on the other modality $Y$, where $Y \in \mathbb{R}^{N\times d_y}$, and $d_y$ is the number of variables. $\Sigma_y^{-1} \cdot V_y^T$ is the Whitening matrix for data $Y$, in the dimension of number of component ($r_y$)-by-number of subject ($N$). $V_y \cdot \Sigma_y$ is the DeWhitening matrix for data $Y$. The top principle components $U_x \in \mathbb{R}^{r_x\times d_x}$ and $U_y \in \mathbb{R}^{r_y\times d_y}$ are input data of ICA.

$$X = \mathrm{V_x} \cdot \Sigma_x \cdot U_x \Rightarrow \Sigma_x^{-1} \cdot V_x^T \cdot X = U_x \tag{3}$$

$$Y = \mathrm{V_y} \cdot \Sigma_y \cdot U_y \Rightarrow \Sigma_y^{-1} \cdot V_y^T \cdot Y = U_y \tag{4}$$

Equation (5) shows that ICA is performed onto two modalities by inputting both $U_x$ and $U_y$ into computation and the output are the mixing matrices $A_x$ and $A_y$. $S_x$ and $S_y$ are independent components. Equation (6) shows the cost function of pICA algorithm which maximizes two entropy terms ($H(Z_x)$ and $H(Z_y)$) and one correlation term ($Corr(A_x, A_y)^2$). The entropy terms are used to maximize the independence of components within each dataset, and the correlation term is used to maximize the connection between components across datasets. The balance between entropy and correlation is dynamically adjusted to prioritize independence, as described in [2].

$$U_x = A_x \cdot S_x; \quad U_y = A_y \cdot S_y \tag{5}$$

$$\begin{aligned}
&\max\left\{H\left(Z_x\right) + H\left(Z_y\right) + \mathrm{Corr}\left(A_x, A_y\right)^2\right\} \\
&= \left\{-E\left[\ln f_z\left(Z_x\right)\right] - E\left[\ln f_z\left(Z_y\right)\right] + \frac{\mathrm{Cov}\left(A_{xi}, A_{yj}\right)^2}{\mathrm{var}\left(A_{xi}\right)\cdot \mathrm{var}\left(A_{yj}\right)}\right\} \\
&Z_x = \frac{1}{1+e^{-K_x}}, \quad K_x = W_x \cdot U_x + W_{x0}, A_x = W_x^{-1} \\
&Z_y = \frac{1}{1+e^{-K_y}}, \quad K_y = W_y \cdot U_y + W_{y0}, A_y = W_y^{-1}
\end{aligned} \tag{6}$$

### C. Decentralized parallel ICA (dpICA)

The goal of dpICA is to achieve the same objectives of pICA under a decentralized environment where raw data are distributed across many sites. dpICA consists of local processing in each local site and the global processing at one site. Local processing involves analyzing raw data, and the global processing analyzes the intermediate results.

The flow of dpICA algorithm is generally based on pICA algorithm and divides PCA into local PCA and global PCA prior performing pICA. While assuming there are two sites as shown in Fig. 1 , each local site performs local PCAs without whitening for datasets X and Y separately. As noted in $X_i$ and $Y_i$ indicate datasets from site I, the output top principal components are $X_i localU$, $Y_i localU$ with dimension of component number $r$-by-variable $d$, the projection matrices are $V_x$ and $V_y$. $X_i localU$, $Y_i localU$ along with their projection matrices from each site are passed to the global site. It is important to note that these U matrices are computed based on a selected number of components from X, Y, so that the original data could not be back-computed.

The global site performs global PCA on the vertically concatenated $U_{XGlobal}$ from all sites as $[X_1 localU^T, ..., X_s localU^T]^T$ and vertically concatenated $U_{YGlobal}$ from all sites as $[Y_1 localU^T, ..., Y_s localU^T]^T$ separately, where $U_{XGlobal}$ and $U_{YGlobal} \in \mathbb{R}^{s\cdot r\times d}$. Each site can have its own specific number of principle components (PCs). Here for illustration purposes, we simplified the formula to set r number of PCs for every site. After individual modalities perform global PCA and whitening, the output of each modality are global principal components $U_{GlobalPCA} \in \mathbb{R}^{r\times d}$ , $Whitening_{Global} \in \mathbb{R}^{r\times s\cdot r_{local}}$ , and $DeWhitening_{Global} \in \mathbb{R}^{s\cdot r_{local}\times r}$ matrices.

The next step of the global site as shown in Fig. 1 is to extract components which maximize the cost function in (6) on global PCA's outputs. This process needs to analyze both modalities in parallel as indicted by the name of the algorithm. Every learning iteration consists of two modules. One is to maximize entropy with respect to W and the other is to maximize correlation with respect to W. The innovation of dpICA is to find local A matrices of all sites to update W in the second module. Finding local A

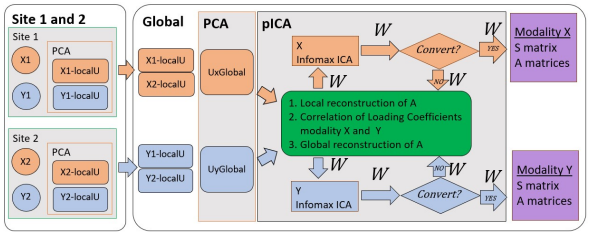



Fig. 1: dpICA algorithm flow.

matrix process is performed on each modality separately. Local A matrices are retrieved based on Equations 7 and 8, which compute the global A matrix ($A_G = W^{-1}$, from the global ICA procedure in the equation 7), and multiply global DeWhitening matrix with global A matrix, followed by multiplication of transposed projection matrices of local sites as in Equation (8). Specifically, horizontally stacked transposed local projection matrices are multiplied in a site-by-site fashion, denoted by $\otimes$, with the product of global DeWhitening and global A matrix. The correlation between $A_L$ matrices of two modalities are then computed and maximized as in the pICA algorithm. Once both local $A_L$

matrices are updated, the updates need to be propagated into $W$ matrices. This is done by updating global $A_G$ matrix and then using the inverse of $A_G$ to update $W$ matrix separately for each modality. As shown in Equation (9), a new global $A_G$ matrix is computed by multiplication of global Whitening matrix with the concatenated products of local projection matrix and the sub-matrix of $A_L$ corresponding to each site. The local projection matrices are multiplied with sub-matrices of $A_L$ in a site-by-site fashion, and then concatenated vertically.

$$U_{GlobalPCA} = A_G \cdot S \tag{7}$$

$$[projection_1^T, projection_2^T, \cdots, projection_s^T,] \otimes \{DeW_{hiteningG} \cdot A_G\} = A_L \tag{8}$$

$$W_{hiteningG} \cdot \left\{ \begin{bmatrix} projection_1 \cdot A_{L1} \\ projection_2 \cdot A_{L2} \\ \cdots \\ projection_s \cdot A_{Ls} \end{bmatrix} \right\} = A_G \tag{9}$$

## III. MATERIALS AND APPLICATIONS

In this section we describe the imaging and genetic data used to test dpICA, including participant information and data preparation, followed by experiments to assess the performance of dpICA.

### A. Participants

Our data coming from Chen et al. [6] consists of MRI imaging and generics from 355 schizophrenia patients and 422 controls. Details about participants can be seen on Demographic Information table in Chen et al. [6].

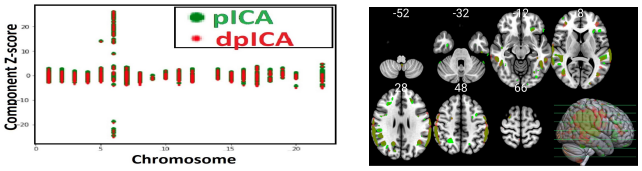
### B. sMRI and Genetic Data

sMRI T1-weighted images were collected from 1.5T/3T scanners, preprocessed by SPM12 (http://www.fil.ion.ucl.ac.uk/spm) to derive 429,655 voxels. Genetic DNA samples are collected from either blood or saliva, genotyped and imputed. 977,242 SNPs were retained after linkage disequilibrium pruning. 1,402 SZ risk SNPs were selected. The details of data preparation is described in Chen et al. [6].

### C. Decentralization Parallel ICA application on sMRI and genetic

In the study done by Chen et al. [6] the pICA analysis extracted 65 components for GMV and 29 for SNP, and identified one SNP component associated with one SZ GMV component. Here, we extended Chen et al's study into decentralized space to identify the same number of components. First we validated dpICA algorithm under one site condition, which is equivalent to the centralized pICA, and then evaluated dpICA performance with multiple experiments under various settings.

The $1^{st}$ dpICA experiment is to analyze the impact of local PCA component number. We performed dpICA with five different settings of local PCA component number using three distributed sites, each site with fixed sample size of 258,259 and 260 respectively. The $2^{nd}$ experiment is to analyze the impact of multi-site condition. We performed dpICA with site numbers of one to five when dividing samples equally across sites. The $3^{rd}$ experiment is to analyze the impact of local sample sizes. We performed dpICA with 14 variations of local sample sizes under 3 site setting. Note each test is conducted for 10 runs and estimated the stable output by ICASSO method [9].




(a) (b)

Fig. 2: Comparison top-pair component between pICA (green) and dpICA (red) of SNP (2a) and sMRI (2b).

## IV. RESULTS

To validate pICA vs. dpICA similarity, we computed correlation coefficient of $A$ and $S$ matrices between algorithms, and compared the the top-pair components and their correlation. As shown in Table I, the comparison of $S$ and $A$ matrices found that the correlation coefficients were high for both modalities. Both pICA and dpICA found the same paired components as illustrated in Fig. 2 and their correlations are also similar. Fig. 2a shows the comparison of pICA (green) and dpICA (red) identified SNP components in chromosome 6. Fig. 2b shows the regions threshold at $|z| > 2$ of the top-paired GMV component of pICA (green) and dpICA (red) plotted by MRIcroGL [10], yellow highlighting the overlapping regions.

Experiment 1: Table II shows that overall top-pair matching is increasing with higher local PCA component numbers. We found that dpICA algorithm performed more stably when the local PCA component number is three times or higher than the global PCA component number. It is indicated by the percent number of matched top-pair component from 10 runs reached to 100%. We call the 3-times rule.

Experiment 2: As demonstrated in Table III , all various multisite settings show the "precent number of matched top-pair component from 10 runs" at 100% and "maximum correlation" at expected range. We found that dpICA algorithm supported multisite when we applied 3-times rule as Experiment 1.

Experiment 3: Table IV 's depiction of the effects of local sample size shows that when 3-times rule is applied, the "percent number of matched top-pair component from 10 runs" is at expected 70%-100% and "maximum correlation" at expected range. This shows that dpICA algorithm supports different local sample size with "3-times rule" condition. Note: The local PCA component number could be set as high as the sample-size number minus one.

## V. DISCUSSION

The findings of our study suggest that dpICA exhibits comparable performance to centralized pICA in facilitating multisite and multimodal data analysis. Notably, we demonstrate that the efficacy of dpICA is contingent upon the condition that the number of local PCA components must be at least three times the number of global PCA components for the optimal performance. This constraint influences both the number of sites and the local sample size. This study utilizes a predetermined number of global PCA components for the purpose of comparison with the previous research. It is worth noting that the optimal number of components can be estimated using information theory based methods, such as the Akaike information criterion or Minimum Description Length.

## VI. CONCLUSIONS

This study shows that pICA can be decentralized with the proposed dpICA algorithm, based on our experiment with sMRI and SNP datasets, which generates outcomes similar to centralized pICA in terms of the maximum correlation between two modalities and the similarity of the independent components and loading matrices of each modality. Under the correct setting of PCA component number, dpICA supports multimodal and multisite analysis. The dpICA algorithm is published on COINSTAC [11].


## ACKNOWLEDGMENT

This project was funded by the National Science Foundation (grant number: 2112455).


## REFERENCES


[1] R. A. Poldrack et al., "Toward open sharing of task-based fMRI data: the OpenfMRI project," Front. Neuroinform., vol. 7, p. 12, Jul. 2013.

[2] J. Liu, G. Pearlson, A. Windemuth, G. Ruano, N. I. Perrone-Bizzozero, and V. Calhoun, "Combining fMRI and SNP data to investigate connections between brain function and genetics using parallel ICA," Hum. Brain Mapp., vol. 30, no. 1, pp. 241–255, Jan. 2009.

[3] J. Liu and V. D. Calhoun, "A review of multivariate analyses in imaging genetics," Front. Neuroinform., vol. 8, p. 29, Mar. 2014.

[4] C. J. Carter, "Schizophrenia susceptibility genes converge on interlinked pathways related to glutamatergic transmission and long-term potentiation, oxidative stress and oligodendrocyte viability," Schizophr. Res., vol. 86, no. 1–3, pp. 1–14, Sep. 2006.

[5] P. J. Harrison and M. J. Owen, "Genes for schizophrenia? Recent findings and their pathophysiological implications," Lancet, vol. 361, no. 9355, pp. 417–419, Feb. 2003.

[6] J. Chen et al., "Shared Genetic Risk of Schizophrenia and Gray Matter Reduction in 6p22.1," Schizophr. Bull., vol. 45, no. 1, pp. 222–232, Jan. 2019.

[7] M. S. Keshavan, R. Tandon, N. N. Boutros, and H. A. Nasrallah, "Schizophrenia, 'just the facts': what we know in 2008 Part 3: neurobiology," Schizophr. Res., vol. 106, no. 2–3, pp. 89–107, Dec. 2008.

[8] J.-F. Cardoso, "Infomax and maximum likelihood for blind source separation," IEEE Signal Process. Lett., vol. 4, no. 4, pp. 112–114, Apr. 1997.

[9] J. Himberg and A. Hyvarinen, "Icasso: software for investigating the reliability of ICA estimates by clustering and visualization," in 2003 IEEE XIII Workshop on Neural Networks for Signal Processing (IEEE Cat. No.03TH8718), Sep. 2003, pp. 259–268.

[10] MRIcroGL, "MRIcroGL 3D Medical Imaging," 2012.

[11] S. M. Plis et al., "COINSTAC: A Privacy Enabled Model and Prototype for Leveraging and Processing Decentralized Brain Imaging Data," Front. Neurosci., vol. 10, p. 365, Aug. 2016.


TABLE I: COMPARISON RESULT BETWEEN PICA AND DPICA

| Validations | Algorithms | |
|---|---|---|
| | piCA | dpICA |
| 1.Maximum correlation | 0.16 | 0.1645 |
| 2.Maximum correlated components | Matched | Matched |
| 3.Similarities of product between pICA and dpICA | Modalities | |
| | sMRI | SNPs |
| 3a.Independent component matrices | $> 0.93$ | $> 0.96$ |
| 3b.Loading matrices of one site | $> 0.93$ | $> 0.96$ |

TABLE II: IMPACT RESULT OF LOCAL PCA COMPONENT NUMBER IN DPICA ALGORITHM.

| No. of site | Sample size per site | Component numbers A/B/C/D[1] | Average Maximum Correlation | % of matched comp. in 10 runs | Ratio of AB vs. CD |
|---|---|---|---|---|---|
| 3 | 258,259,260 | 65/29/65/29 | 0.1398 | 90 | 1 |
| 3 | 258,259,260 | 130/58/65/29 | 0.1827 | 70 | 2 |
| 3 | 258,259,260 | 195/87/65/29 | 0.1866 | **100** | **3** |
| 3 | 258,259,260 | 257/116/65/29 | 0.1832 | **100** | **4** |
| 3 | 258,259,260 | 257/145/65/29 | 0.1812 | **100** | **5** |

[1] [A/B/C/D]; when A is local sMRI, B is local SNP, C is global sMRI, D is global SNP.

TABLE III: IMPACT RESULT OF N-SITE IN DPICA ALGORITHM.

| No. of site | Sample size per site | Component numbers A/B/C/D[1] | Average Maximum Correlation | % of matched comp. in 10 runs |
|---|---|---|---|---|
| 1 | 777 | 257/116/65/29 | 0.2556 | **100** |
| 2 | 388,389 | 257/116/65/29 | 0.1711 | **100** |
| 3 | 258,259,260 | 257/116/65/29 | 0.1756 | **100** |
| 4 | 194,194,194,195 | 193/116/65/29 | 0.1701 | **100** |
| 5 | 155,155,155,155,157 | 154/145/65/29 | 0.1775 | **100** |

TABLE IV: IMPACT RESULT OF LOCAL SAMPLE SIZE IN DPICA ALGORITHM.

| No. of site | Sample size per site | Component numbers A/B/C/D[1] | Average Maximum Correlation | % of matched comp. in 10 runs | Sample distribution method |
|---|---|---|---|---|---|
| 3 | 196,196,385 | 195/87/65/29 | 0.2443 | 80 | Not Equally |
| 3 | 385,196,196 | 195/87/65/29 | 0.1854 | **100** | Not Equally |
| 3 | 196,385,196 | 195/87/65/29 | 0.26 | **100** | Not Equally |
| 3 | 196,260,321 | 195/87/65/29 | 0.1858 | 80 | Not Equally |
| 3 | 260,196,321 | 195/87/65/29 | 0.1827 | **100** | Not Equally |
| 3 | 196,321,260 | 195/87/65/29 | 0.1798 | 80 | Not Equally |
| 3 | 260,321,196 | 195/87/65/29 | 0.1808 | 90 | Not Equally |
| 3 | 321,196,260 | 195/87/65/29 | 0.1738 | **100** | Not Equally |
| 3 | 321,260,196 | 195/87/65/29 | 0.1763 | **100** | Not Equally |
| 3 | 260,260,257 | 195/87/65/29 | 0.182 | 70 | Not Equally |
| 3 | 260,257,260 | 195/87/65/29 | 0.2468 | 80 | Not Equally |
| 3 | 257,261,259 | 195/87/65/29 | 0.1843 | 70 | Not Equally |
| 3 | 258,259,260 | 257/116/65/29 | 0.1799 | **100** | Equally |
| 3 | 258,259,260 | 257/145/65/29 | 0.1764 | **100** | Equally |