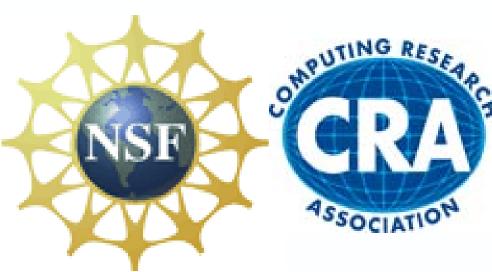
Graphical User Interface for Kidney Paired Donation Program



Yanhua Chen, Peter X.K. Song {chenyanh, pxsong}@umich.edu **Department of Biostatistics** University of Michigan, Ann Arbor, MI 48109, USA





Introduction

A graphical user interface (GUI) to visualize inputs and outputs from the Kidney Paired Donation (KPD) program is developed. KPD program is based primarily on micro-simulation models which enable us to evaluate and compare allocation strategies and effects of policy. There is no user friendly platform thus far that permits easy communications between inputs and outputs in the KPD program. Thus, the developed interface will facilitate the clinical research related to KPD program.

KPD Program and Optimization

I. Background

- □ 80,000 people awaiting kidney transplants, 16,000 transplants performed yearly. The availability of organs does not meet the excessive demand
- □ Donors incompatible with their recipients can still help their loved ones get live donor kidneys by matching with other incompatible pairs

2. Objective

☐ Make **optimal** decisions to reach the **most mutual benefits** (e.g., more patients save life) from limited **resources** (kidney donors)

3. Solution: Optimized Matches via Constrained Graphic Optimization

- ☐ Graph: Each **node** represents an incompatible donor/recipient pair or altruistic donor, each **edge** represents a possible match
- □ Match: (1) Paired Donor-Patient Exchanges: $3 \Leftrightarrow 5$, $2 \Leftrightarrow 4 \Leftrightarrow 6$; (2) Chains for Altruistic Donors: 7 = > 1; (3) Combination of (1) and (2)

Altruistic donor (1, 0.1)

Algorithm

Formulation: Integer Programming

$$\max \sum_{c \in C} y_c u_c,$$

$$s.t., y_c \in \{0,1\}, \forall c \in C,$$

$$and \sum_{c \in C(i)} y_c \leq 1, 1 \leq i \leq n$$

where C is the class of all cycles or chains of length three or less without or with altruistic donors, C(i) is the set of cycles or chains in C that contain vertex i and y_c is a vector of indicators representing if cycle or chain c is to be executed for transplant $(y_c = 1)$ or not $(y_c = 0)$, u_c is the expected utility of cycle c depending on edge utility eii and probability pii.

eriod=1. Id=39. Type=Pair. Blood=O. Gene=DR8.DR11.DR12.DR13.DR14.DR17.DR18

Period=1, Id=43, Type=Pair, Blood=0, Gene=DR8, DR11, DR12, DR13, DR14, DR17, DR18,

Period=2, Id=51, Type=Pair, Blood=0, Gene=DR8,DR11,DR12,DR13,DR14,DR17,DR18,

Period=2, Id=66, Type=Pair, Blood=0, Gene=A1,B15,CW4,D06,DR13,DR52,B37,CW6,D02,DR7,DR53,BV Period=2, Id=67, Type=Pair, Blood=0, Gene=A1,B35,CW12,DQ6,DR15,DR51,B37,CW6,BW4,BW6, Period=2, Id=68, Type=Pair, Blood=A, Gene=A1,B39,CW12,D05,DR16,DR51,B38,CW6,D08,DR4,DR53,BW

Period=2, Id=69, Type=Pair, Blood=A, Gene=A1,B44,CW5,DO5,DR14,DR52,B39,CW12,DR16,DR51,BW4,BW

Period=3, Id=75, Type=Pair, Blood=O, Gene=A1,B40,CW10,DQ4,DR8,DR53,B51,CW1,DQ5,DR1,BW4,BW

Period=3, Id=77, Type=Pair, Blood=0, Gene=A1,B18,CW12,D05,DR10,B35,CW6,D07,DR11,DR52,BW6,

Period=3, Id=81, Type=Pair, Blood=A, Gene=A1,B27,CW2,DQ8,DR4,DR53,B40,DQ5,DR14,DR52,BW6,

Period=3, Id=82, Type=Pair, Blood=B, Gene=A1,B18,CW7,DQ7,DR11,DR52,B38,CW12,DQ2,DR17,BW6

Period=3, Id=76, Type=Pair, Blood=B, Gene=A1,B27,CW15,DQ2,DR7,DR53,B18,CW6,DQ7,DR11,DR52,BW

Period=3, Id=78, Type=Pair, Blood=0, Gene=A1,B44,CW7,DQ6,DR15,DR51,B38,CW12,DQ7,DR4,DR53,BW4,BW6

Period=3, Id=80, Type=Pair, Blood=O, Gene=A1,B39,CW12,DQ7,DR11,DR52,B45,CW6,DQ2,DR7,DR53,BW4,BW6.

Period=2, Id=70, Type=Pair, Blood=0, Gene=A1,B40,CW2,D07,DR11,DR52,B35,CW4,BW4,BW6, Period=2, Id=71, Type=AD, Blood=O, Gene=A1,B35,CW4,DQ6,DR15,DR51,B18,CW7,DQ7,DR103,BW6 Period=3, Id=72, Type=Pair, Blood=A, Gene=A1,B38,CW6,DQ8,DR4,DR53,B40,CW10,DQ2,DR7,BW4,BW6

Period=3, Id=73, Type=Pair, Blood=A, Gene=A1,B27,CW15,DO2,DR7,DR53,B51,DO7,DR4,BW6,

Period=3, Id=79, Type=Pair, Blood=A, Gene=A1,B38,CW6,DQ7,DR11,DR52,B51,CW1,BW4,BW6,

Period=5, BeginVertexId=80, EndVertexId=83, EdgeUtility=10.000000, EdgeProbability=0.40175

Period=5, BeginVertexId=80, EndVertexId=88, EdgeUtility=10.000000, EdgeProbability=0.455187

Period=5, BeginVertexId=80, EndVertexId=22, EdgeUtility=10.000000, EdgeProbability=0.204678 Period=5, BeginVertexId=80, EndVertexId=23, EdgeUtility=10.000000, EdgeProbability=0.168667

Period=5, BeginVertexId=80, EndVertexId=26, EdgeUtility=10.000000, EdgeProbability=0.459252 Period=5, BeginVertexId=80, EndVertexId=41, EdgeUtility=10.000000, EdgeProbability=0.28451

Period=5, BeginVertexId=80, EndVertexId=47, EdgeUtility=10.000000, EdgeProbability=0.285125

Period=5, BeginVertexId=80, EndVertexId=54, EdgeUtility=10.000000, EdgeProbability=0.11063 Period=5, BeginVertexId=80, EndVertexId=94, EdgeUtility=10.000000, EdgeProbability=0.347163

Period=5, BeginVertexId=80, EndVertexId=97, EdgeUtility=10.000000, EdgeProbability=0.208390 eriod=5, BeginVertexId=80, EndVertexId=114, EdgeUtility=10.000000, EdgeProbability=0.152138° Period=5, BeginVertexId=80, EndVertexId=115, EdgeUtility=10.000000, EdgeProbability=0.253362

Period=5, BeginVertexId=80, EndVertexId=116, EdgeUtility=10.000000, EdgeProbability=0.113208 Period=5, BeginVertexId=82, EndVertexId=14, EdgeUtility=10.000000, EdgeProbability=0.107007 Period=5, BeginVertexId=82, EndVertexId=16, EdgeUtility=10.000000, EdgeProbability=0.31007

eriod=5, BeginVertexId=82, EndVertexId=23, EdgeUtility=10.000000, EdgeProbability=0.154449

eriod=5, BeginVertexId=82, EndVertexId=54, EdgeUtility=10.000000, EdgeProbability=0.483874

Period=3, Id=74, Type=Pair, Blood=A, Gene=A1,B35,CW4,D05,DR14,DR52,B39,CW12,BW6,

Period=1, Id=45, Type=Pair, Blood=A, Gene=A2,A68,A69,B46,CW2,CW5,CW6,CW8,CW14,CW18,DR7,DR8,DR9,DR11,DR12,DR13,DR15,DR1

Period=1, Id=46, Type=Pair, Blood=0, Gene=A23,A24,A25,A29,A32,B7,B13,B27,B2708,B37,B38,B4005,B44,B45,B47,B48,B49,B51,B52,B53,B

Period=2, Id=50, Type=Pair, Blood=0, Gene=A23,A24,A3001,A3101,A32,B7,B13,B27,B2708,B35,B37,B38,B39,B4005,B41,B44,B45,B46,B47,

Period=2, Id=52, Type=Pair, Blood=0, Gene=A2,A23,A24,A25,A26,A29,A31,A32,A33,A34,A36,A43,A66,A6601,A6602,A68,A69,A74,A80,B4

Period=2 Td=55 Type=Pair Blood=0 Gene=∆6802 B8 B18 B35 B37 B38 B39 B41 B42 B5102 B53 B54 B55 B59 B64 B65 B67 B71 B73 B75 B7

Period=1, Id=40, Type=Pair, Blood=0, Gene=DQ5,DQ6,

Period=1, Id=47, Type=Pair, Blood=0, Gene=D05,D06,

Period=2, Id=54, Type=Pair, Blood=B, Gene=A1102,

Period=1, Id=41, Type=Pair, Blood=0, Gene=DQ5,DQ6

The solution to the above constrained maximization provides the optimal kidney exchange in a KPD program

124 Pair

Cycle-With-AD

Cycle-With-AD

Cycle-With-AD

Total 5 number of lab match run(s) summary

Lab match run#1: 2 matches, 20.000000 utilitie

Lab match run#2: 1 matches, 10.000000 utilities

Lab match run#3: 0 matches, 0.000000 utilities

Lab match run#4: 3 matches, 30,000000 utilities

Lab match run#5: 5 matches, 50.000000 utilities

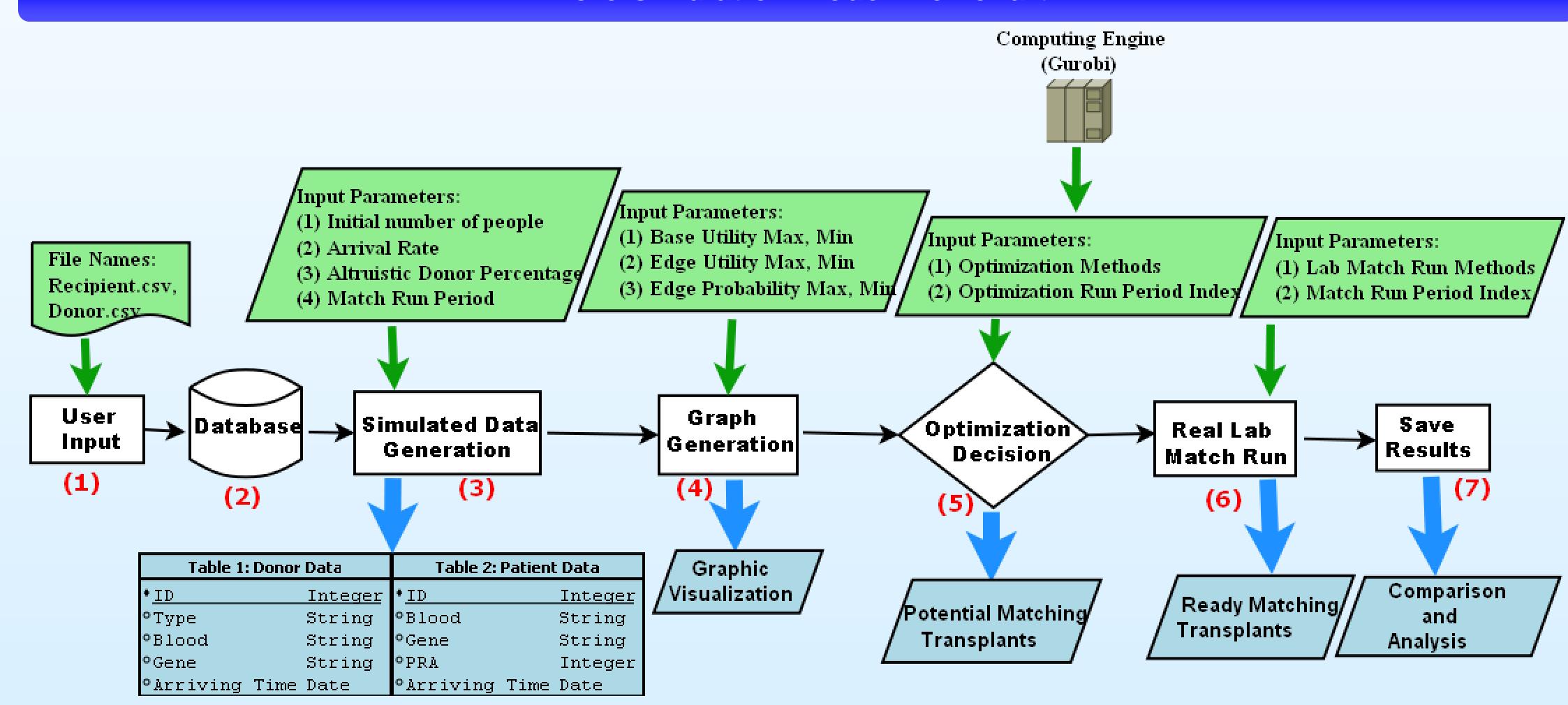
Fotal successful transplant(s) utility is: 110.000000

Total successful transplant(s) number is: 11

System Configuration

- 1. The software consists of a comprehensive *frontend* GUI in Qt development environment and a backend computing engine powered by optimization integer programming software, such as Gurobi.
- □ Computing Engine: the core of KPD program applies statistical data models and optimization functions to output matching results. It is written entirely in C++ and compiled into a binary executable code.
- □GUI's Primary Role: receive and process data from users, select modeling criteria and KPD parameters, and then to output results in easily accessible environments. GUI is developed in C++ too and also compiled into a separate executable program.
- 2. Communication between the computing engine and the GUI is carried out through input and output files. Thus, the KPD software can be easily deployed across multiple platforms, such as Windows, Linux and Mac OS.

Micro-simulation Model Flowchart



GUI Software

Basic Components:

- Read data from users
- 2. Build (secured) database
- Generate simulation data from database
- 4. Generate graphic visualization
- 5. Optimization decision: output potential matching transplants
- 6. Real lab match run: output ready matching transplants
- Output and save results for analysis and comparison

Major Operations:

Step 1: Choose menu: *File -> Open ->* Input file names into *Read KPD Recipient and Donor File* dialog Step 2: Select menu: Application -> KPD Data Generation -> Assign parameters in KPD Data Generation Parameters dialog -> Push OK to confirm the parameters -> Output the simulated recipients and donors data in the Recipients window and **Donors** window

Step 3: Select menu: Application -> KPD Graph Generation -> Assign parameters in KPD Graph Generation Parameters dialog -> Push OK to confirm the parameters -> Output the vertex and edge of simulated graph in the **Graph Build** window

Step 4: Select menu: Application -> KPD Optimization Run -> Choose method and period in KPD Optimization Run *Input* dialog -> Push *OK* to confirm -> Output the potential matching results in the *Central* window Step 5: Select menu: Application -> KPD Lab Match Run -> Choose method and period in KPD Match Run Input

dialog -> Push OK to confirm -> Output the ready matching results in the Central window Step 6: Select menu: *File -> Save* -> Assign file name (.csv) for the current result -> Push *save* button to confirm

Future Work

- 1. The system will allow to input data of patient/donor pairs and/or altruistic donors from three sources:
 - ☐ Existing secured database: BMTR (Bone Marrow Transplant Registry), APD (Allied for Paired Donation), NKR (National Kidney Registry), SRTR (Scientific Registry of Transplant Recipients), Stanford (Stanford Medical School Blood Center), UM (University of Michigan KPD Program).
 - ☐ Statistical models that are trained from simulated data from the existing secured database
 - ☐ New inputs of patient/donor and/or altruistic donor data directly from users
- 2. The system will build graphic visualization tools into the existing software for output analysis, which helps clinicians, donors and patients more easily visualize and assess the KPD program.

Acknowledgements

- 1. This research was funded by the US National Science Foundation NSF), Computing Research Association(CRA), and Computing Community Consortium (CCC) under grants No. CIF (Computing Innovation Fellows)-B-66 (2010-2011).
- 2. Thanks to Professor Jack D. Kalbfleisch, Yijiang Li and Yan Zhou of Department of Biostatistics, University of Michigan.