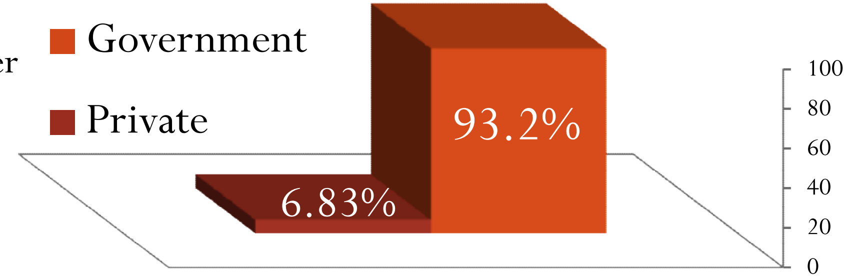
THE PROBLEM OF EFFICIENT BLOOD TRANSFUSION

- Esra Aleisa – Kuwait University

- In this research, refers to the problem of efficiently collecting and distributing blood products in an environment containing stochastic supply (donation) of blood products and stochastic demand for blood products.

NATIONAL CENTRAL BLOOD BANK

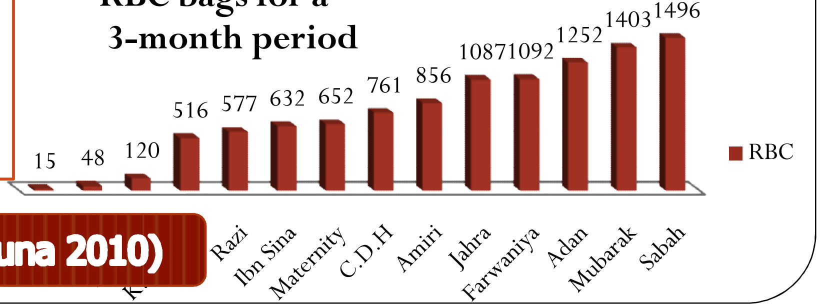
- Collects blood from donors, process it, store it, and deliver it to public and private hospitals.
- Donors are not paid
- Public hospitals get it for free.
- Private hospitals pay for blood.



Demand for RBC

OBJECTIVE

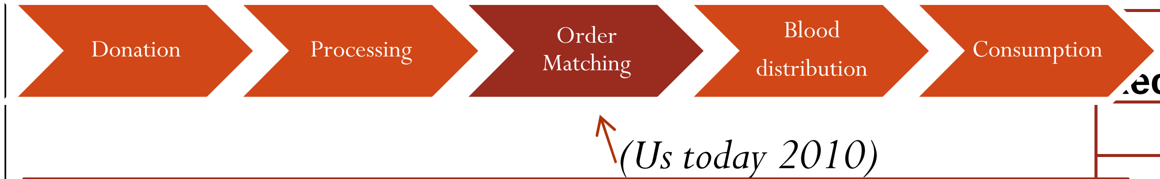
- Deliver the best possible type and quantity of blood requirement while minimizing shortages and waste on RBC.



Kuwait's blood bank receives 60,000 donors annually (Kuna 2010)

BLOOD SUPPLY AND SIMULATION

4 ECHELON SYSTEM *(Spens & Bask, 2002)*



PREVIOUS WORK

- Most work in this area is in blood-bank screening and management of patients (Bern et al. 2008, Pierskalla, Kitchen & P. L. Chiodini, 2006)
- Early related work in in the area of storage in perishable goods (Nahmias and Pierskalla, 1973, Pierskalla et al, 1977; Prastacos, 1984).
- Platelet inventory (Blake, 2003)
- Cross matching (Jagannathan and Sen, 1991)
- Blood distribution study (Wells et al 2002)
- RFID for blood supply chain (Asif & Mandviwalla 2005, Bednarz, 2004)

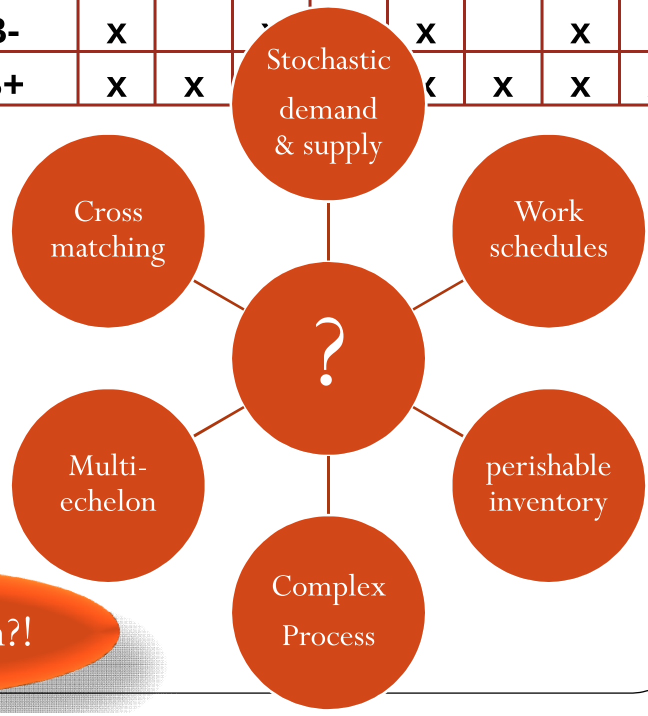
CLOSELY RELATED WORK

- **Simulation of blood distribution network/ inventory**

  Katsaliaki and Brailsford 2006, Spens & Bask 2002, Rytilä and Spens 2007, Dijk 2005, Haijemaet et al 2004)

BLOOD CROSS MATCHING

	Donor							
Recipient	O-	O+	A-	A+	B-	B+	AB-	AB+
O-	x							
O+	x	x						
A-	x		x					
A+	x	x	x	x				
B-	x				x			
B+	x	x			x	x		
AB-	x		x	[illegible]	x		x	
AB+	x	x	[illegible]	[illegible]	x	x	x	x



Optimization?!

SIMULATION & OPTIMIZATION - BLOOD

In Haijema, Van der Wal and Van Dijk (2004)

STEPS

combined OR and simulation by the following steps:

Step 1: First, a stochastic dynamic programming

- **STEP 1:** Provide a Stochastic Dynamic Programming (SDP) formulation
- **STEP 2:** Reduce the dimension of the (SDP) by aggregating the state space and demands so that the downsized (SDP) problem can be solved numerically (using successive approximation).
- **STEP 3:** The obtained optimal solution (policy) is (re)evaluated and run by simulation.
- **STEP 4:** a heuristic search procedure used derive a simple practical near to optimal solution.
- **STEP 5:** The quality of this simple order-up-to strategy is then evaluated by simulation.

Due to the complexity of problem- The demand and inventory levels are downsized (aggregate pools in batches of 4)– obtain optimal solution for the downsized problem

SDP

The state of the system is described by ***(d,x)*** with:

d: the day of the week (d = 1,2,…,7)

x = (x1, x2,…, xm) the inventory state

x_r = *the* number of pools with a residual life time of *r days* (maximal *m = 6 days*)

<u>A pool is one patient transfusion</u>

Vn(d,x): represent the minimal expected costs over n days when starting in state *(d,x).*

The optimal inventory strategy is determined by solving for the SDP-equations for *n* = *1,2,…..*

$$V_n(d,x) = \min_k \left[c(x,k) + \sum p_d(b)\, V_{n-1}(d+1, t(x,k,b)) \right]$$

k the production action

c(x,k) cost per day in state x under production k

$p_d(b)$ the probability for a composite demand d

t(x,k,b) the new inventory state based on k, b, x

BLOOD CROSS MATCHING

Assignment Problem ?

THE ASSIGNMENT PROBLEM

FROM WIKIPEDIA

- The assignment problem is one of the fundamental combinatorial optimization problems in the branch of optimization or operations research in mathematics. It consists of finding a maximum weight matching in a weighted bipartite graph.

- **THE ASSIGNMENT PROBLEM**

Let there be n jobs which are to be assigned to n operators so that one job is assigned to only one operator.

i = Index for job, $i = 1, 2, \ldots n$

j = Index for operators, $j = 1, 2, \ldots n$

C_{ij} = Unit cost for assigning job i to operator j

$$X_{ij} = \begin{cases} 1 \text{ if job i is assigned to operator j} \\ 0 \text{ Otherwise} \end{cases}$$

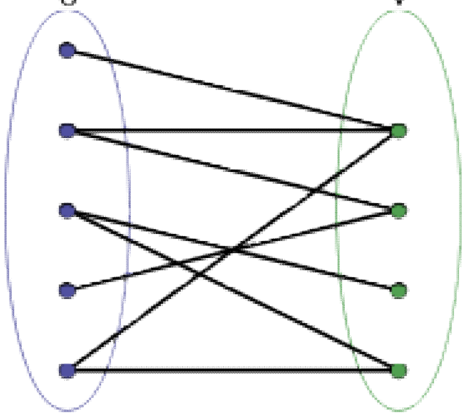
The objective function is:

$$Minimize\ Z = \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij}$$

$$Subject\ to \quad \sum_{i=1}^{n} X_{ij} = 1; for\ all\ j; j = 1, 2, \ldots N$$

$$\sum_{j=1}^{n} X_{ij} = 1; for\ all\ i; i = 1, 2, \ldots N$$

$$X_{ij} \geq for\ all\ i\ and\ all\ j.$$



There are a number of agents and a number of tasks. Any agent can be assigned to perform any task, incurring some cost that may vary depending on the agent-task assignment. It is required to perform all tasks by assigning exactly one agent to each task in such a way that the total cost of the assignment is minimized.

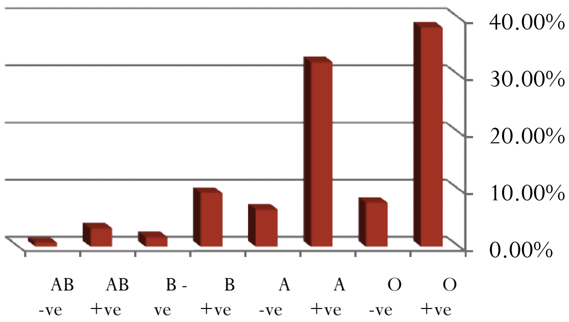
Benanav et al 1985 shows that this type of problem is NP- complete

ORDER FULFILLMENT

- **OPERATORS AND TASKS**
- Operators: Different blood type inventories
- Tasks: Blood type demands regardless of hospital

$c_{ij} = \begin{cases} 1 \text{ if type i is compatible with type j} \\ 0 \text{ Otherwise} \end{cases}$

$X_{rk} = \begin{cases} 1 \text{ if sack r will be assigned to order k} \\ 0 \text{ Otherwise} \end{cases}$



D_{ik} = Demand for type i for hospital k

S_i = Available storage for type i

P_i = Available pooled inventory for type i

$Pi = \sum_{i=1}^{n} c_{ij} S_i$ *for all i*

Ti= Time sack k been in storage

- *For each order k, route through each sacks r in pool i*
  - *Rank pool according to remaining perishability 1= spent more time in inventory*
    - *Break ties* $1 / Si$
    - *Exact match of blood type*
  - *Update rank*
  - *Assign* X_{rk}
- *Repeat*

Objective: minimize shortage

Cross match to pull from nearest expiring inventory

Cross match to pull from fullest inventory

Rarer blood types (slow moving) have higher priority

Smaller orders given higher priority

TYPES	DISTRIBUTION	RATIOS
O +ve	1 person in 3	38.4%
O -ve	1 person in 15	7.7%
A +ve	1 person in 3	32.3%
A -ve	1 person in 16	6.5%
B +ve	1 person in 12	9.4%
B -ve	1 person in 67	1.7%
AB +ve	1 person in 29	3.2%
AB -ve	1 person in 167	0.7%

http://www.bdwebguide.com

Include any other complex subjective selection rules

Develop an embedded optimizer within the simulation model