Imperial College London

Computer Networks and G-Networks

Erol Gelenbe

www.ee.imperial.ac.uk/gelenbe

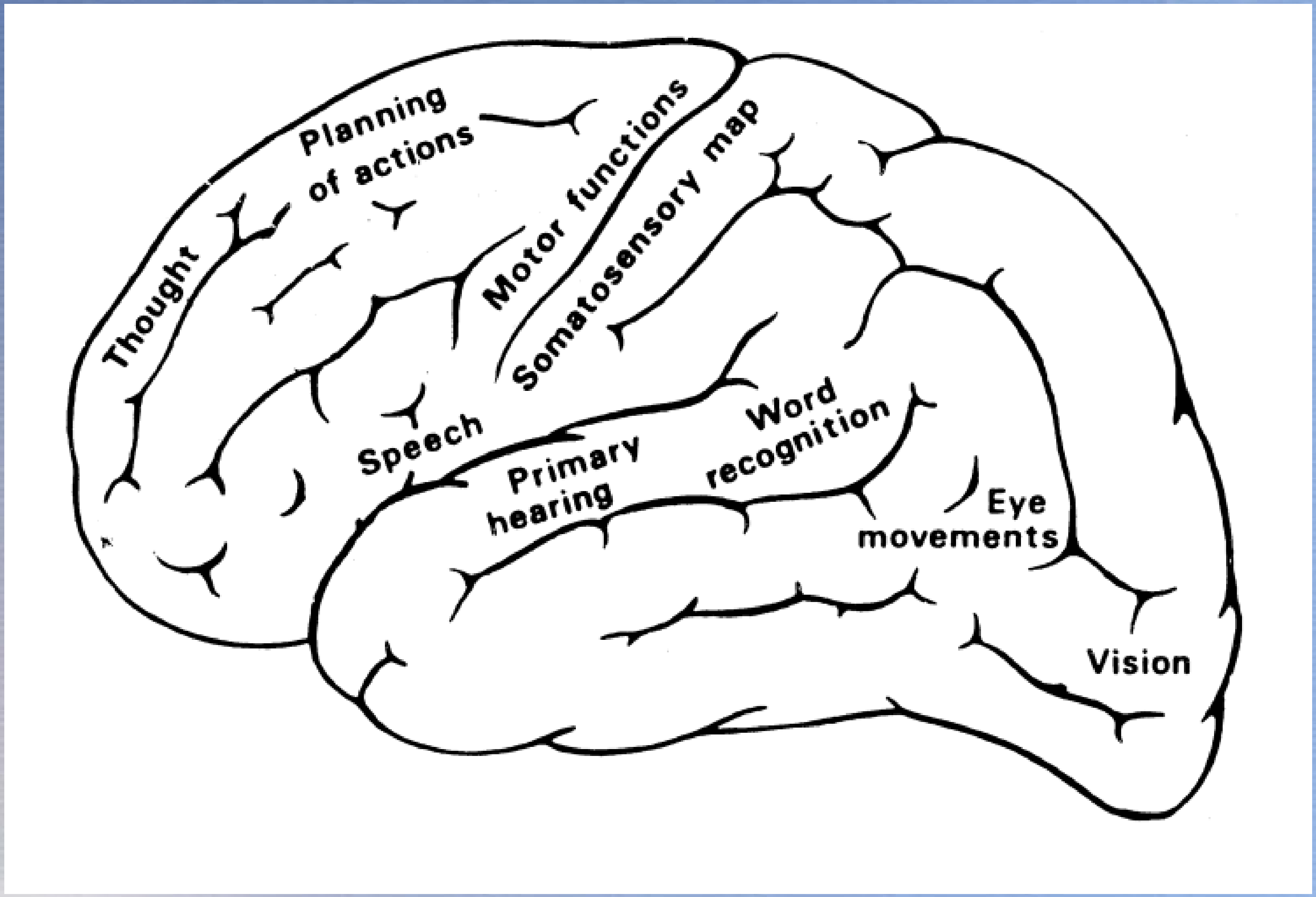
Dept of Electrical and Electronic Engineering
Imperial College
London SW7 2BT

Imperial College London

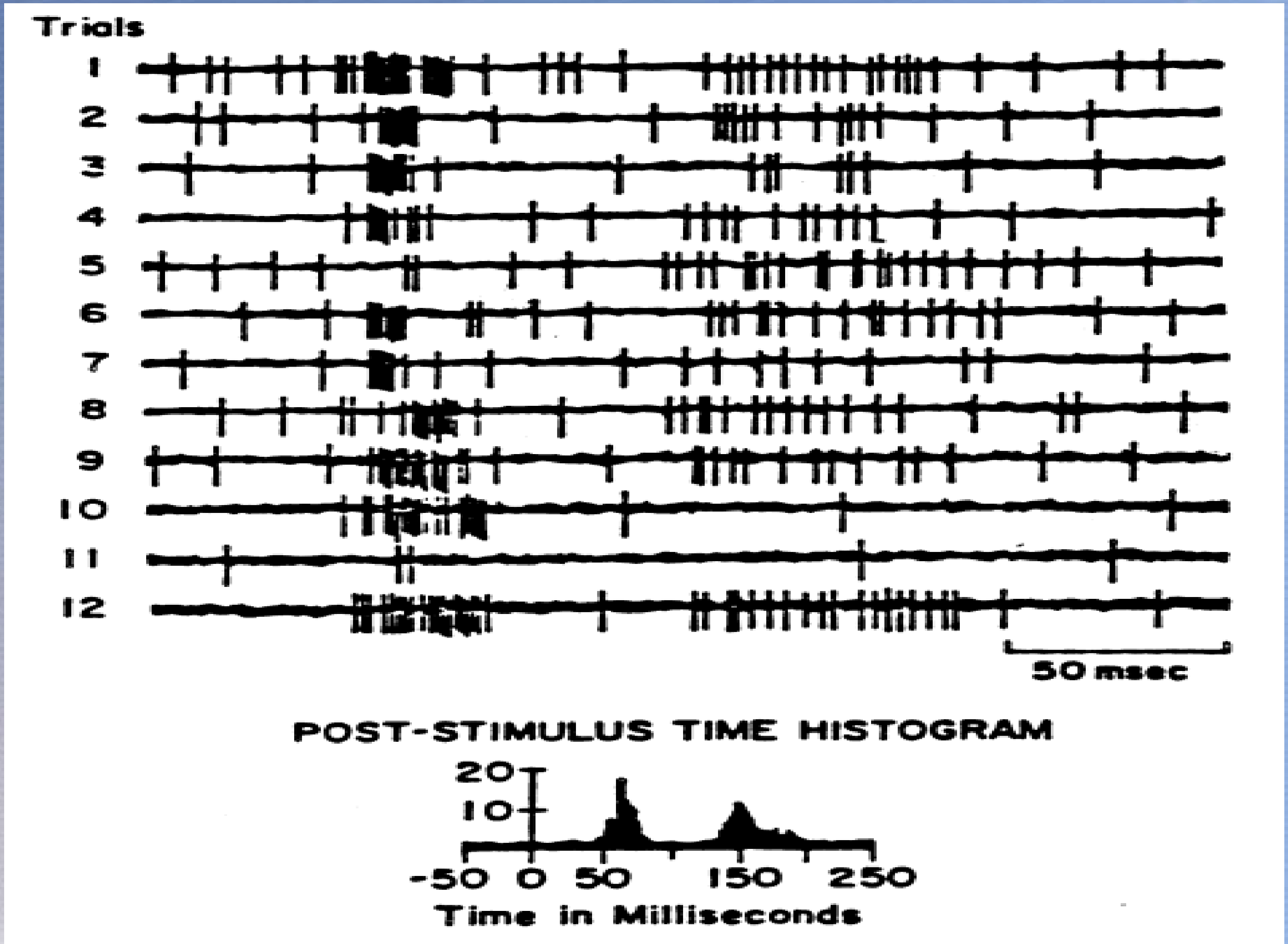
Outline

- Origin of G-Networks: the RNN
- Biological Inspiration for the RNN
- Applications
  - modeling biological neuronal systems
  - texture recognition and segmentation
  - image and video compression
  - multicast routing
  - Network routing (Cognitive Packet Network)
- Gene Regulatory Systems
- Networked Economics: Auctions

Random Spiking Behaviour of Neurons



Random Spiking Behaviour of Neurons



Imperial College London

The RNN: A Model of Random Spiking Neurons

Some biological characteristics that the model should include:

- Action potential “Signals” in the form of spikes
- Excitation-inhibition spikes
- Modeling recurrent networks
- Random delays between spikes
- Conveying information along axons via variable spike rates
- Store and fire behaviour of the soma
- Reduction of neuronal potential after firing
- Possibility of representing axonal delays between neurons
- Arbitrary network topology
- Ability to incorporate different learning algorithms: Hebbian, Gradient Descent, Reinforcement Learning, ..
- Synchronised firing patterns
- Logic in neural networks?

Queuing Networks + Stochastic Petri Nets : Exploiting the Analogy

Discrete state space, typically continuous time, stochastic models arising in the study of populations, dams, production systems, communication networks ..

- Theoretical foundation for computer and network systems performance analysis
- Open (external Arrivals and Departures), as in Telephony, or Closed (Finite Population) as in Compartment Models
- Systems comprised of Customers and Servers
- Theory is over 100 years old and still very active ..
- Big activity at Telecom labs in Europe and the USA, Bell Labs, AT&T Labs, IBM Research
- More than 100,000 papers on the subject ..

Queuing Network <-> Random Neural Network

- Both Open and Closed Systems
- Systems comprised of Customers and Servers
- Servers = Neurons
- Customer = Spike: Arriving to server will increase the queue length by +1
- Excitatory spike arriving to neuron will increase its soma's potential by +1
- Service completion (neuron firing) at server (neuron) will send out a customer (spike), and reduce queue length by 1
- Inhibitory spike arriving to neuron will decrease its soma's potential by 1
- Spikes (customers) leaving neuron i (server i) will move to neuron j (server j) in a probabilistic manner

RNN

Mathematical properties that we have established:

o Product form solution

o Existence and uniqueness of solution and closed form analytical solutions for arbitrarily large systems in terms of rational functions of first degree polynomials

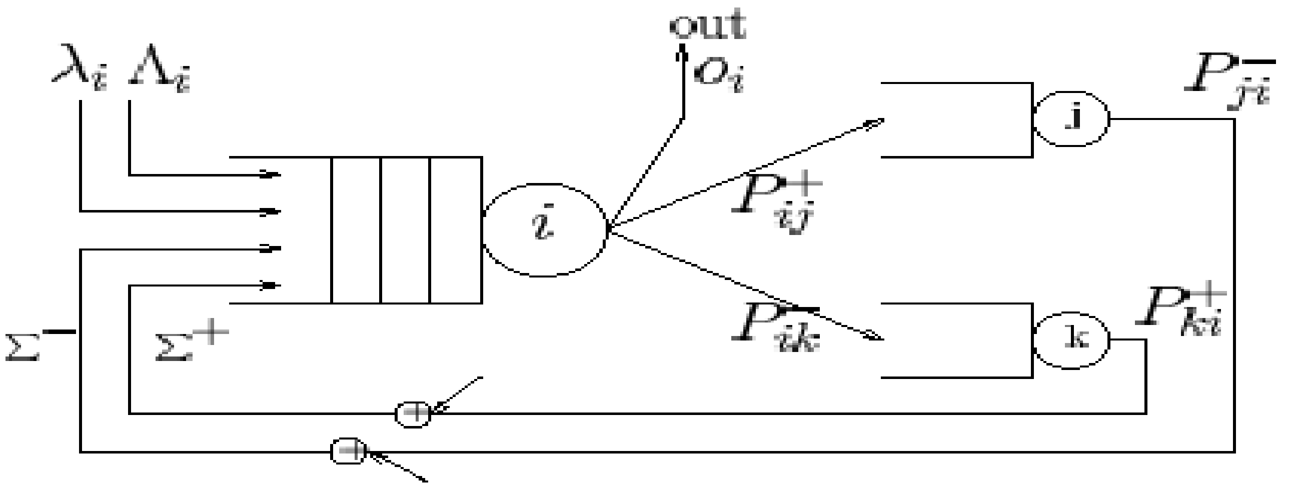
o Strong inhibition – inhibitory spikes reduce the potential to zero

o The feed-forward RNN is a universal computing element: for any bounded continuous function f: $R^n \to R^m$, and an error ε, there is a FF-RNN g such that $||g(x)-f(x)|| < \varepsilon$ for all x in R^n

o $O(n^3)$ speed for recurrent network's gradient descent algorithm, and $O(n^2)$ for feedforward network

Imperial College London

The RNN



- This is a spiked neural network model .. excitation spikes "+1" and inhibition spikes "-1" travel in the newtork
- The state of neuron i is a non-negative integer k_i
- The state of the n-neuron network is a vector $(k_1, ..., k_n)$

Mathematical Model: A "neural" network with n neurons

- Internal State of Neuron i at time t, is an Integer $K_i(t) \geq 0$
- Network State at time t is a Vector

$$K(t) = (K_1(t), \ldots , K_i(t), \ldots , K_k(t), \ldots , K_n(t))$$

- If $K_i(t) > 0$, we say that Neuron i is excited it may fire (in which case it will send out a spike)
- Also, if $K_i(t) > 0$, the Neuron i will fire with probability $r_i \Delta t + o(\Delta t)$ in the interval $[t, t+\Delta t]$
- If $K_i(t) = 0$, the Neuron cannot fire at t^+

When Neuron i fires at time t:

- It sends a spike to some Neuron j, with probability p_{ij}
- Its internal state changes $K_i(t^+) = K_i(t) - 1$

Imperial College London

Mathematical Model: A "neural" network with n neurons

The arriving spike at Neuron j is an:

- Excitatory Spike w.p. p_{ij}^{+}
- Inhibitory Spike w.p. p_{ij}^{-}
- $p_{ij} = p_{ij}^{+} + p_{ij}^{-}$ with $\Sigma^{n}_{j=1} p_{ij} \leq 1$ for all i=1,..,n

From Neuron i to Neuron j:

- Excitatory Weight or Rate is $w_{ij}^{+} = r_i \, p_{ij}^{+}$
- Inhibitory Weight or Rate is $w_{ij}^{-} = r_i \, p_{ij}^{-}$
- Total Firing Rate is $r_i = \Sigma^{n}_{j=1} (w_{ij}^{+} + w_{ij}^{-})$

To Neuron i, from Outside the Network

- External Excitatory Spikes arrive at rate Λ_i
- External Inhibitory Spikes arrive at rate λ_i

Imperial College London

State Equations & Their Solution

$p(k,t) = \Pr[x(t)=k]$ where $\{x(t):t \geq 0\}$ is a discrete state-space Markov process,

and $k_{ij}^{+-} = k + e_i - e_j$, $k_{ij}^{++} = k + e_i + e_j$

$k_i^{+} = k + e_i$, $k_i^{-} = k - e_i$:

The **Chapman - Kolmogorov** Equations [Neural Master Equations]:

$$\frac{d}{dt} p(k,t) = \sum_{i,j} [p(k_{ij}^{+-},t) r_i p_{ij}^{+} 1[k_j(t)>0] + p(k_{ij}^{++},t) r_i p_{ij}^{-}] + \sum_i [p(k_i^{+},t)(\lambda_i + r_i d_i) + \Lambda_i p(k_i^{-},t) 1[k_i(t)>0]]$$

$$- p(k,t) \sum_i [(\lambda_i + r_i) 1[k_i(t)>0] + \Lambda_i]$$

Let :

$$p(k) = \lim_{t\to\infty} \Pr[x(t)=k], \quad \textit{and} \quad q_i = \lim_{t\to\infty} \Pr[x_i(t)>0]$$

Theorem : [Gelenbe *Neural Computation* **'90]** If the C-K equations have a stationary solution, then the solution has the "product form" $p(k) = \prod_{i=1}^{n} q_i^{k_i}(1-q_i)$, where

Imperial College London

$$0 \leq q_i = \frac{\Lambda_i + \sum_j q_j r_j \overbrace{p^{+}_{ji}}^{\omega^{+}_{ji}}}{r_i + \lambda_i + \sum_j q_j r_j \underbrace{p^{-}_{ji}}_{\omega^{-}_{ji}}} < 1$$

Probability that Neuron *i* is excited

External Arrival Rate of Excitatory Spikes

ω^{+}_{ji}

Firing Rate of Neuron *i*

External Arrival Rate of Inhibitory Spikes

ω^{-}_{ji}

Theorem (Gelenbe *Neural Computation* '93)

The system of non-linear equations

$$q_i = \frac{\Lambda_i + \sum_j q_j r_j p^+_{ji}}{r_i + \lambda_i + \sum_j q_j r_j p^-_{ji}} \quad , \quad 1 \le i \le n$$

has an unique solution if all the $q_i < 1$.

Theorem (Gelenbe, Mao, Da-Li *IEEE Trans. Neural Nets.* '99)

Let $g : [0,1]^v \to R$ *be continuous and bounded. For any* $\varepsilon > 0$, *there exists an RNN with two output neurons* q_{o+}, q_{o-} *s.t.*

$$\sup_{x \in [0,1]^v} | g(x) - y(x) | < \varepsilon \quad \text{for} \quad y(x) = \frac{q_{o+}}{1 - q_{o+}} - \frac{q_{o-}}{1 - q_{o-}}$$

Random Neural Network

- Neurons exchange Excitatory and Inhibitory Spikes (Signals)
- Inter-neuronal Weights are Replaced by Firing Rates
- Neuron Excitation Probabilities obtained from Non-Linear State Equations
- Steady-State Probability is Product of Marginal Probabilities
- Separability of the Stationary Solution based on Neuron Excitation Probabilities
- Existence and Uniqueness of Solutions for Recurrent Network
- Learning Algorithms for Recurrent Network are $O(n^3)$
- Multiple Classes (1998) and Multiple Class Learning (2002)

Imperial College London

Sample of Publications

- E. Gelenbe. Random neural networks with negative and positive signals and product form solution. *Neural Computation*, 2:239-247, Feburary 1990.
- E. Gelenbe. Learning in the recurrent random neural network. *Neural Computation*, 5:154-164, 1993.
- E. Gelenbe and C. Cramer. Oscillatory corthico-thalamic response to somatosensory input. *Biosystems*, 48(1-3):67-75, November 1998.
- E. Gelenbe and J.M. Fourneau. Random neural networks with multiple classes of signals. *Neural Computation*, 11(4):953-963, May 1999.
- E. Gelenbe, Z.H. Mao, and Y.D. Li. Function approximation with spiked random networks. *IEEE Transactitons on Neural Networks*, 10(1):3-9, January 1999.
- E. Gelenbe and K. Hussain. Learning in the multiple class random neural network. *IEEE Transactions on Neural Networks*, 13(6):1257-1267, November 2002.
- E. Gelenbe, T. Koçak, and Rong Wang. Wafer surface reconstruction from top-down scanning electron microscope images. *Microelectronic Engineering*, 75(2):216-233, August 2004.
- E. Gelenbe, Z.H. Mao, and Y.D. Li. Function approximation by random neural networks with a bounded number of layers. *Journal of Differential Equations and Dynamical Systems*, 12(1-2):143-170, 2004.
- E. Gelenbe, S. Timotheou. Random Neural Networks with Synchronised Interactions. *Neural Computation. 2008*

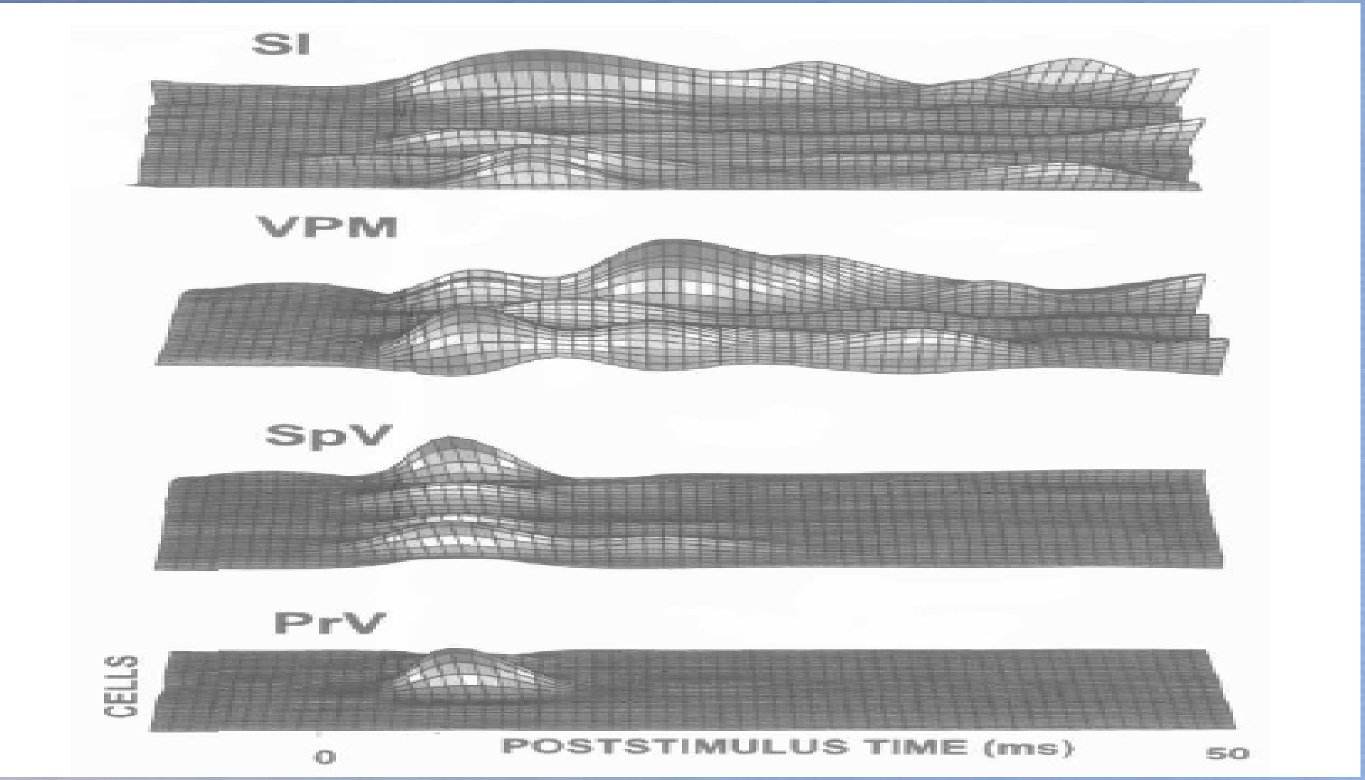
Imperial College
London

Some Applications

- Cortico-Thalamic Response …
- Texture based Image Segmentation
- Image and Video Compression
- Multicast Routing
- CPN Routing

Imperial College
London

Cortico-Thalamic Oscillatory Response to Somato-Sensory Input
(what does the rat think when you tweak her/his whisker?)



Input from the brain stem (PrV) and response at thalamus (VPM) and cortex (SI), reprinted from M.A.L. Nicollelis et al. “Reconstructing the engram: simultaneous, multiple site, many single neuron recordings”, *Neuron* vol. 18, 529-537, 1997

Imperial College London

Scientific Objective
Elucidate Aspects of Observed Brain Oscillations

- Clarify Some of the Mechanisms which Influence (Brain) Cortico-Thalamic Oscillations

- Start with Oscillations Observed in a Physiologically well understood system: the Rat "Barrel Neurons"

- Use a Recurrent "Random Network (RNN)" Spiked Model which actually Models the (Observed) Natural Neurons' Spiked Behaviour

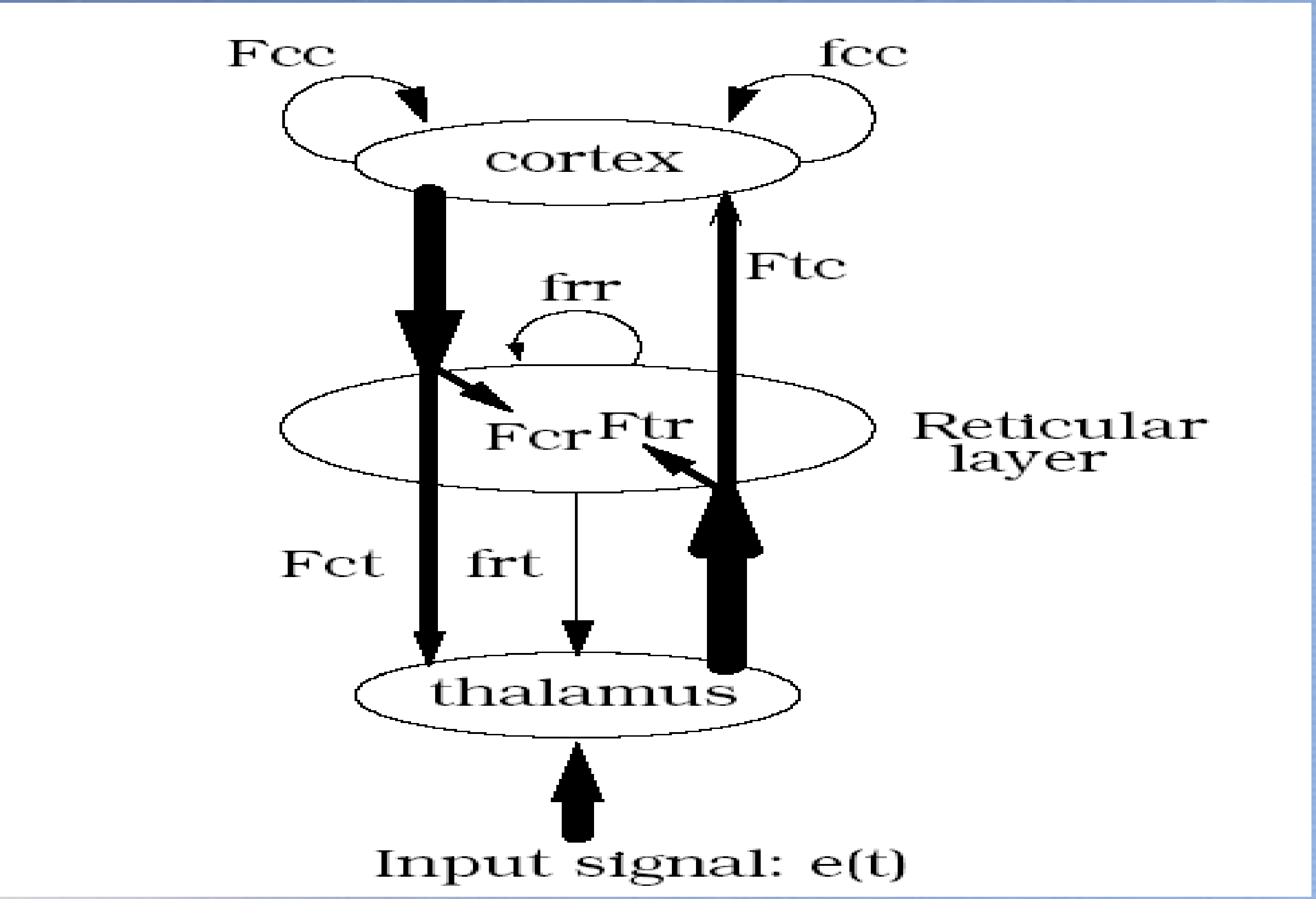
- Identify Primary Factors Causing Oscillations

Imperial College
London

The Biological Model

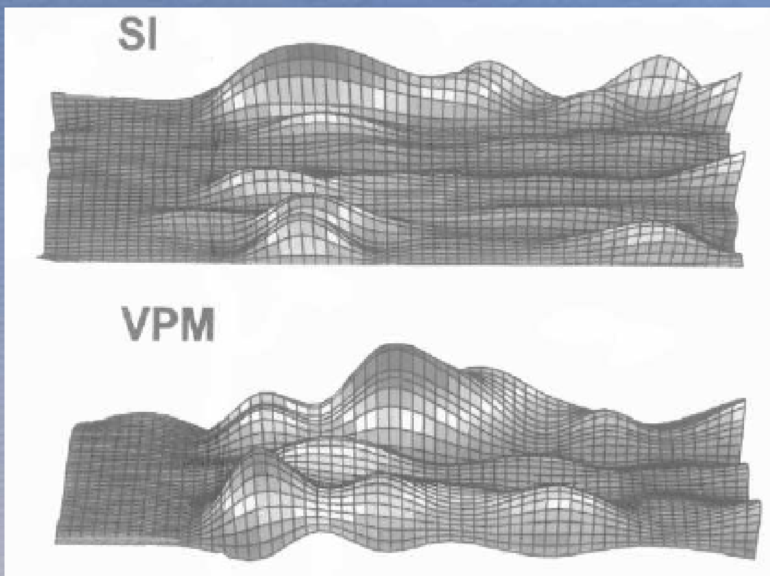
- The somatosensory stimulus (involving a single whisker of the rat) impacts a physiological system in which the number of thalamic cells T is of the order of 10^3, while 10^2 cortical cells C are involved.
- The model assumes that all cortical cells involved are statistically identical, that all thalamic cells are statistically identical, and that all reticular layer cells are also statistically identical.
- In relation to Simons et al., thalamic cells T correspond to thalamo-cortical units (TCU), cortical cells C correspond to "regular spike" barrel units (RSU) ofsomatosensory cortex.

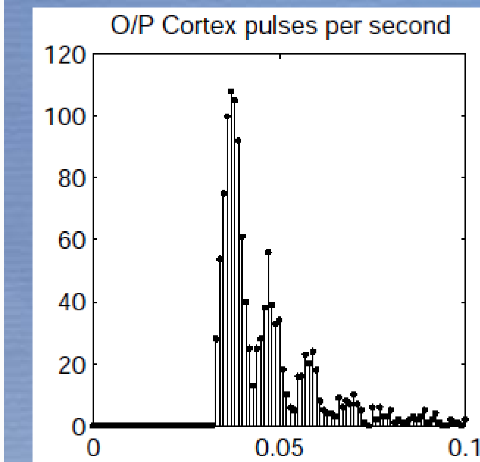
Building the Network Architecture from Physiological Data

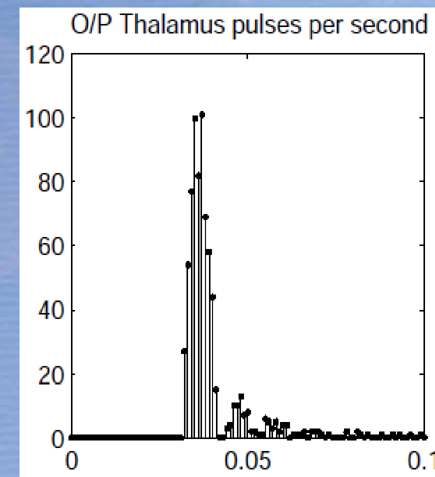


Imperial College London

First Step: Comparing Measurements and Theory: Calibrated RNN Model and Cortico-Thalamic Oscillations





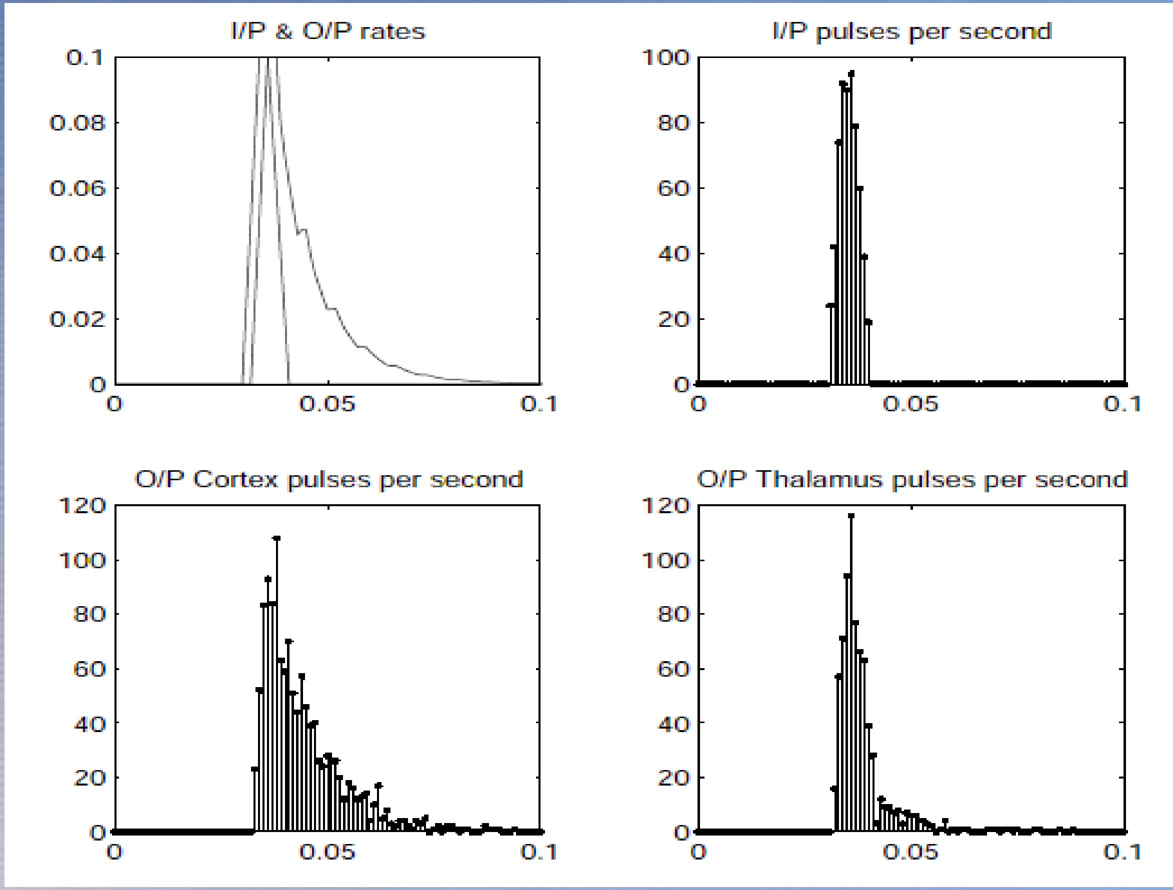


Simultaneous Multiple Cell Recordings (Nicollelis et al., 1997)

Predictions of Calibrated RNN Mathematical Model (Gelenbe & Cramer ’98, ’99)

Imperial College London

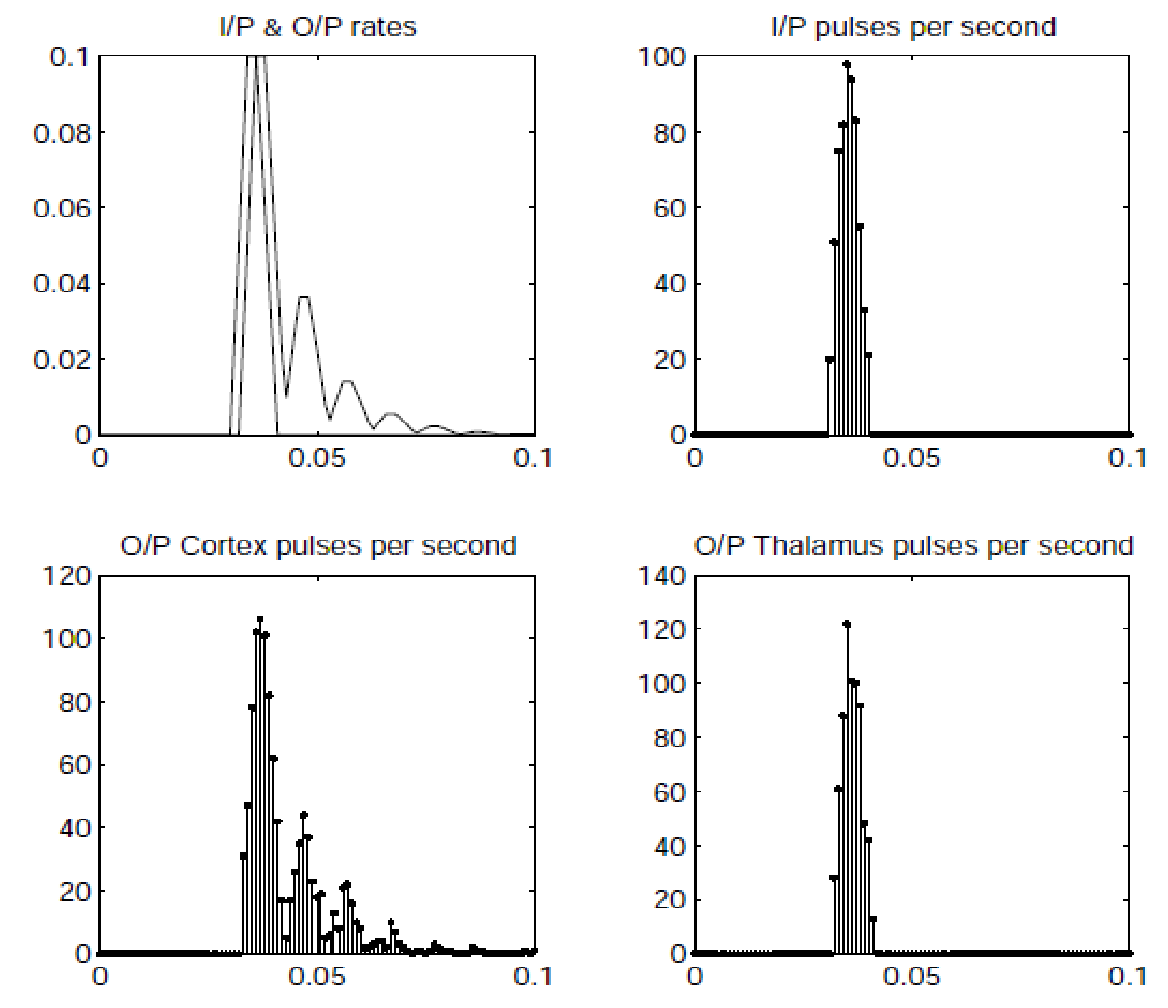
Gedanken Experiments that cannot be Conducted in Vivo: Oscillations Disappear when Signaling Delay in Cortex is Decreased



Brain Stem Input Pulse Rate

Imperial College London

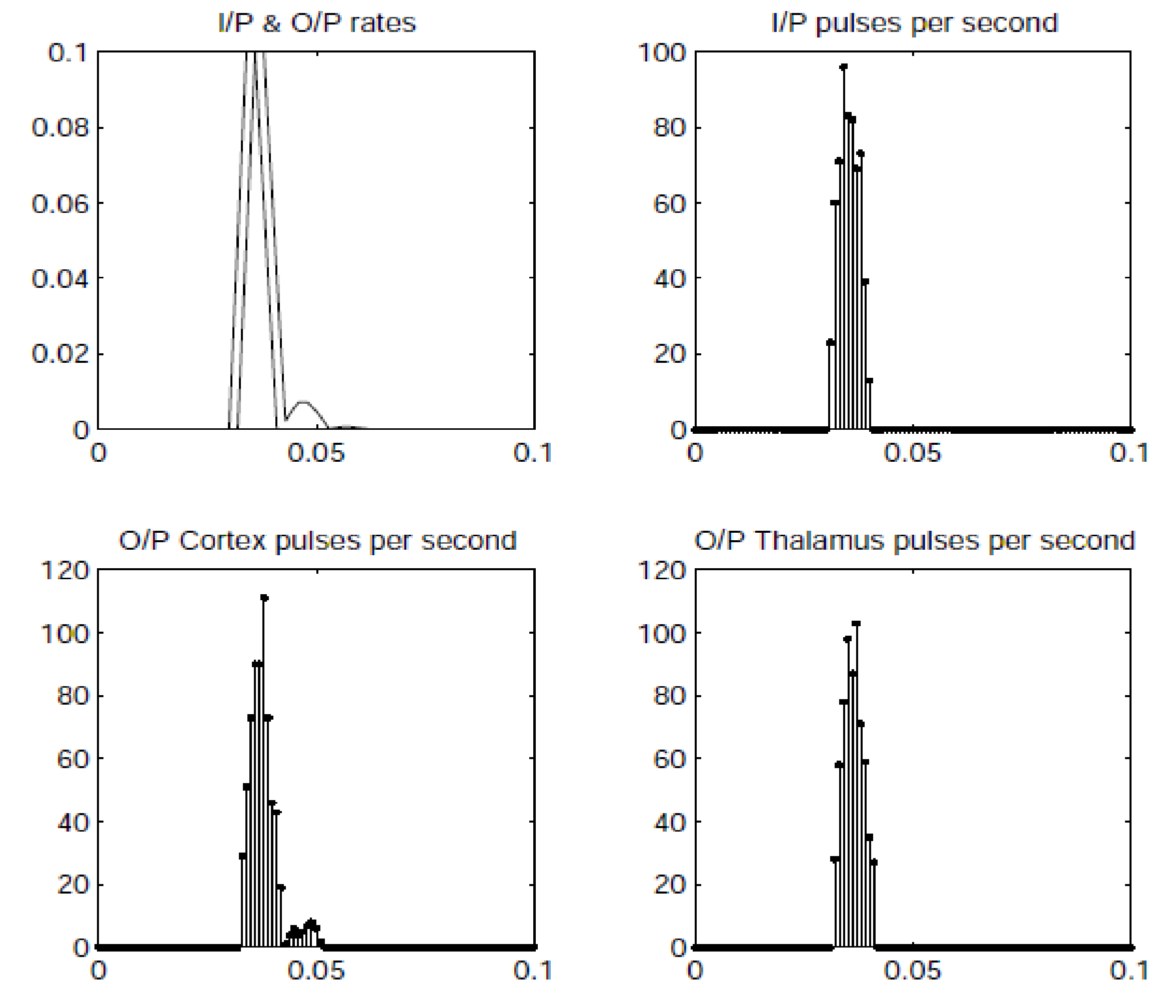
Gedanken Experiments: Removing Positive Feedback in Cortex Eliminates Oscillations in the Thalamus



Brain Stem Input Pulse Rate

Imperial College London

When Feedback in Cortex is Dominantly Negative, Cortico-Thalamic Oscillations Disappear Altogether



Brain Stem Input Pulse Rate

Imperial College London

Summary of Findings Resulting from the Model

- Positive Feedback loops within cortex significantly affect the existence of the damped oscillatory phenomenon, and its duration.
- Positive cortex to thalamus Feedback is not needed for cortical oscillations, but is needed for thalamic oscillations.
- Cortex to thalamus negative Feedback via the reticular layer, and cortex to cortex inhibitory connections, contribute to damping.
- The reticular layer affects the amplitude of the oscillations, but not their causes. Projections from thalamus to cortex reduce the amplitude but do not modify the damping constants or periods of the oscillations.

Imperial College London

On to Some CS/EE Applications of the Random Neural Network

Imperial College London

Building a Practical "Learning" Algorithm: Gradient Computation for the Recurrent RNN is $O(n^3)$

Let $\mathbf{q} = (q_1, \cdots, q_n)$, and define the $n \times n$ matrix

$$\mathbf{W} = \{[w^+(i,j) - w^-(i,j)q_j]/\lambda^-(j)\} \quad i,j = 1, \cdots, n.$$

The vector equations can now be written as

$$\partial \mathbf{q}/\partial w^+(u,v) = \partial \mathbf{q}/\partial w^+(u,v)\mathbf{W} + \gamma^+(u,v)q_u$$
$$\partial \mathbf{q}/\partial w^-(u,v) = \partial \mathbf{q}/\partial w^-(u,v)\mathbf{W} + \gamma^-(u,v)q_u$$

where the elements of the n vectors

$$\gamma^+(u,v) = [\gamma_1^+(u,v), \cdots, \gamma_n^+(u,v)]$$

and $\gamma^-(u,v) = [\gamma_1^-(u,v), \cdots, \gamma_n^-(u,v)]$ are

$$\gamma_i^+(u,v) = \begin{cases} -1/\lambda^-(i) & \text{if } u = i, v \neq i \\ +1/\lambda^-(i) & \text{if } u \neq i, v = i \\ 0 & \text{otherwise} \end{cases}$$

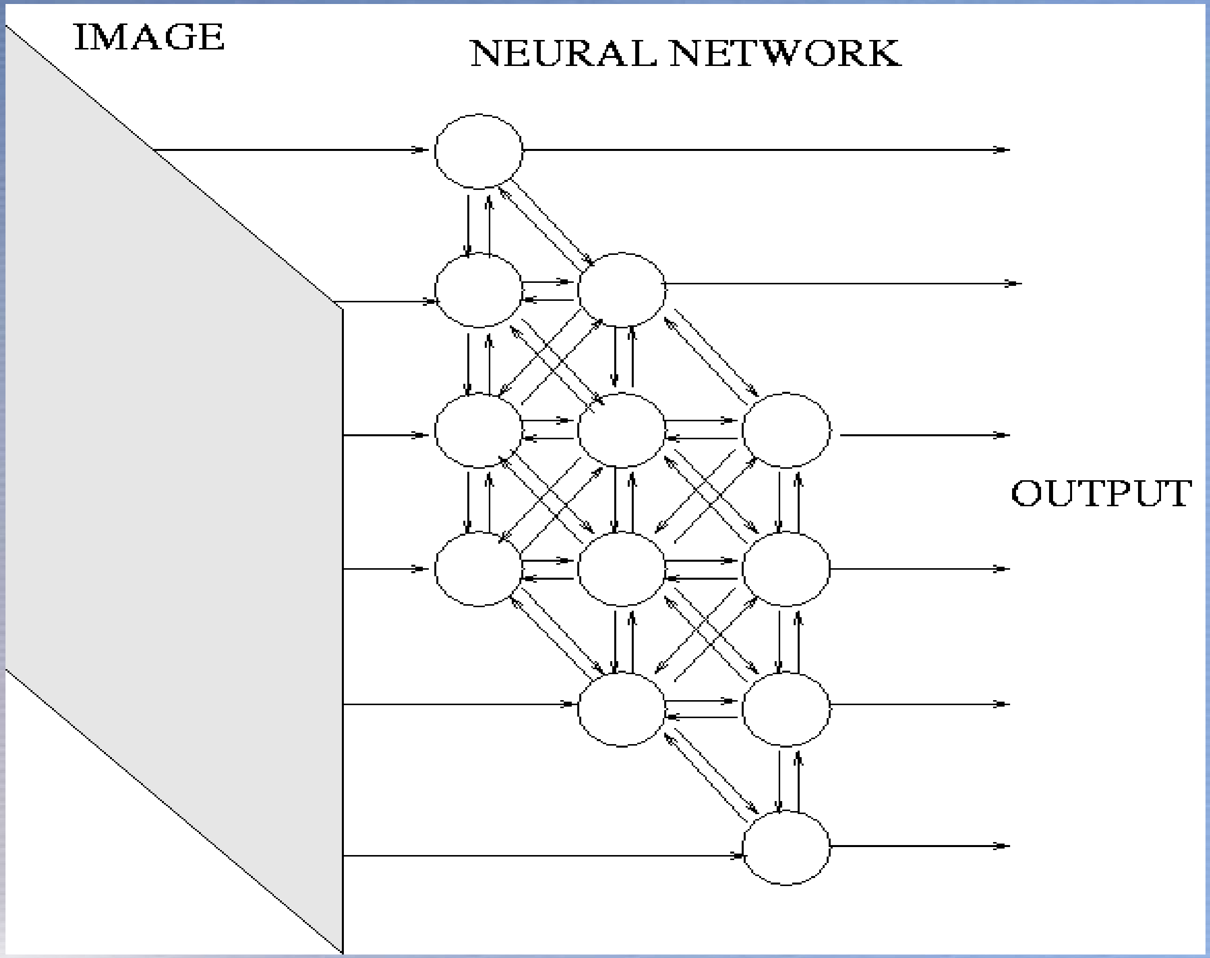
$$\gamma_i^-(u,v) = \begin{cases} -(1+q_i)/\lambda^-(i) & \text{if } u = i, v = i \\ -1/\lambda^-(i) & \text{if } u = i, v \neq i \\ -q_i/\lambda^-(i) & \text{if } u \neq i, v = i \\ 0 & \text{otherwise.} \end{cases}$$

Notice that

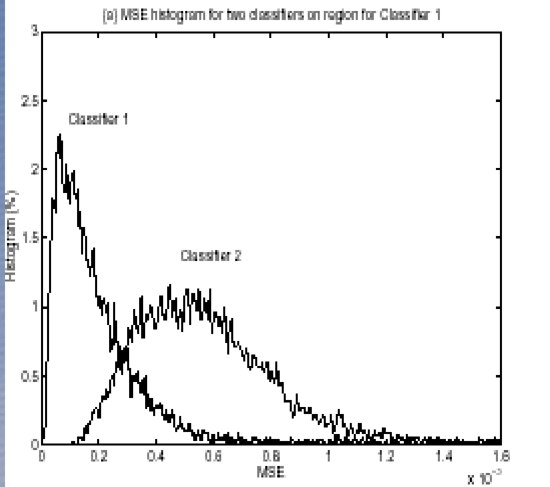
$$\begin{aligned} \partial \mathbf{q}/\partial w^+(u,v) &= \gamma^+(u,v)q_u[\mathbf{I} - \mathbf{W}]^{-1} \\ \partial \mathbf{q}/\partial w^-(u,v) &= \gamma^-(u,v)q_u[\mathbf{I} - \mathbf{W}]^{-1} \end{aligned} \qquad (5)$$

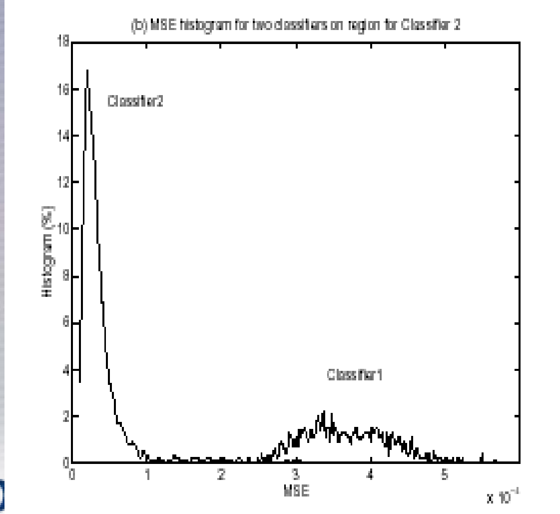
where $\mathbf{I}$ denotes the $n \times n$ identity matrix. Hence the main computational effort in this algorithm is to obtain $[\mathbf{I} - \mathbf{W}]^{-1}$. This is of time complexity $O(n^3)$ or $O(mn^2)$ if an m-step relaxation method is used.

Texture Based Object Identification Using the RNN US Patent '99 (E. Gelenbe, Y. Feng)



1) MRI Image Segmentation





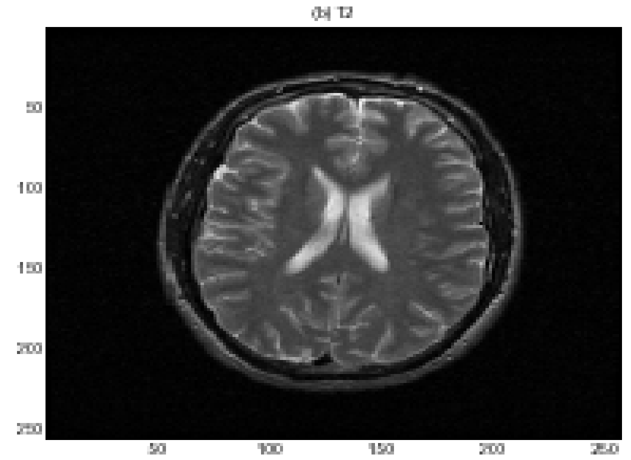


Figure 7: A T2 MR image before being processed by our method

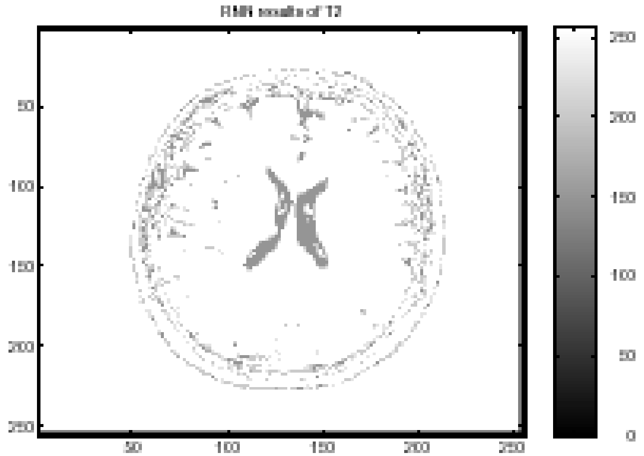
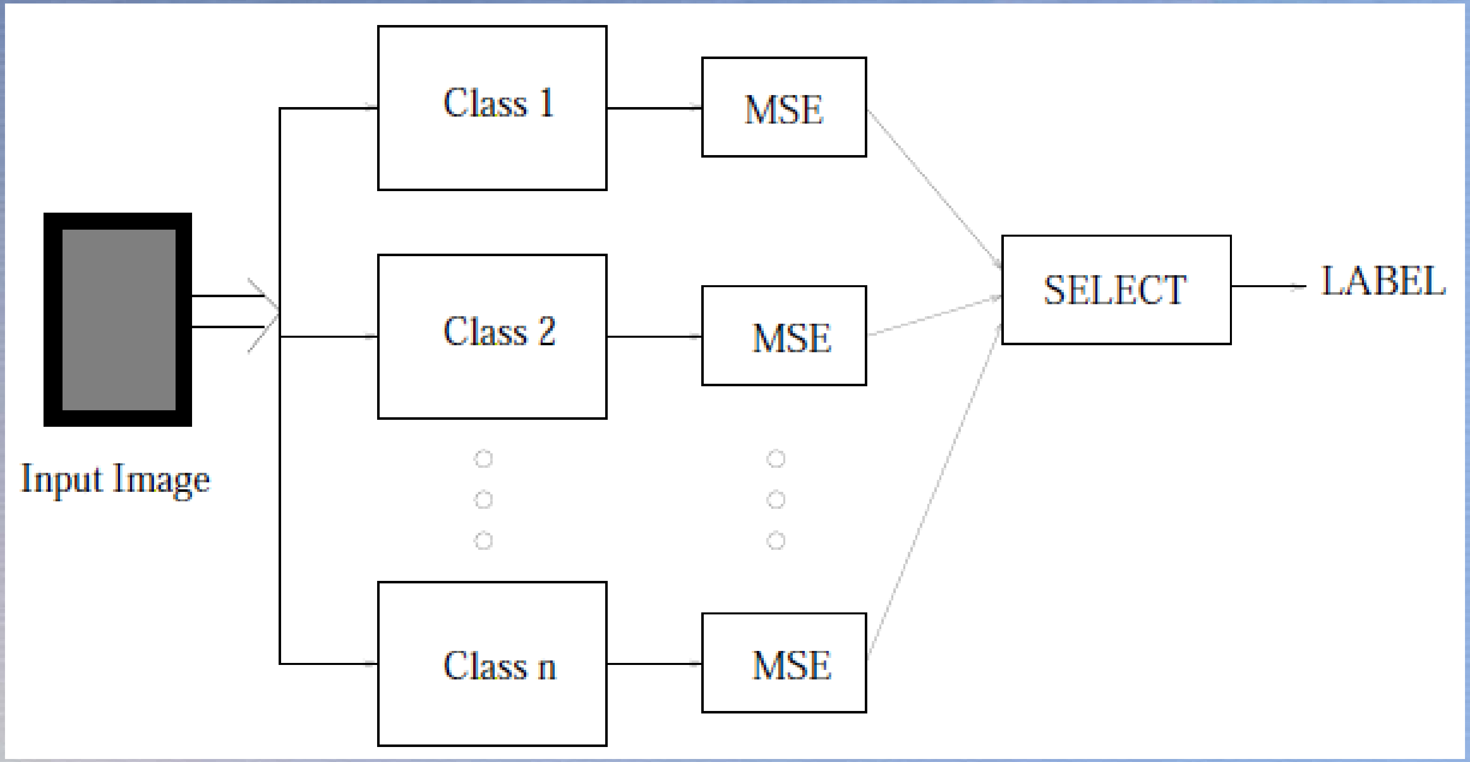


Figure 8: Result of RNN processing for the T2 MRI image.

MRI Image Segmentation



Imperial College London

Brain Image Segmentation with RNN

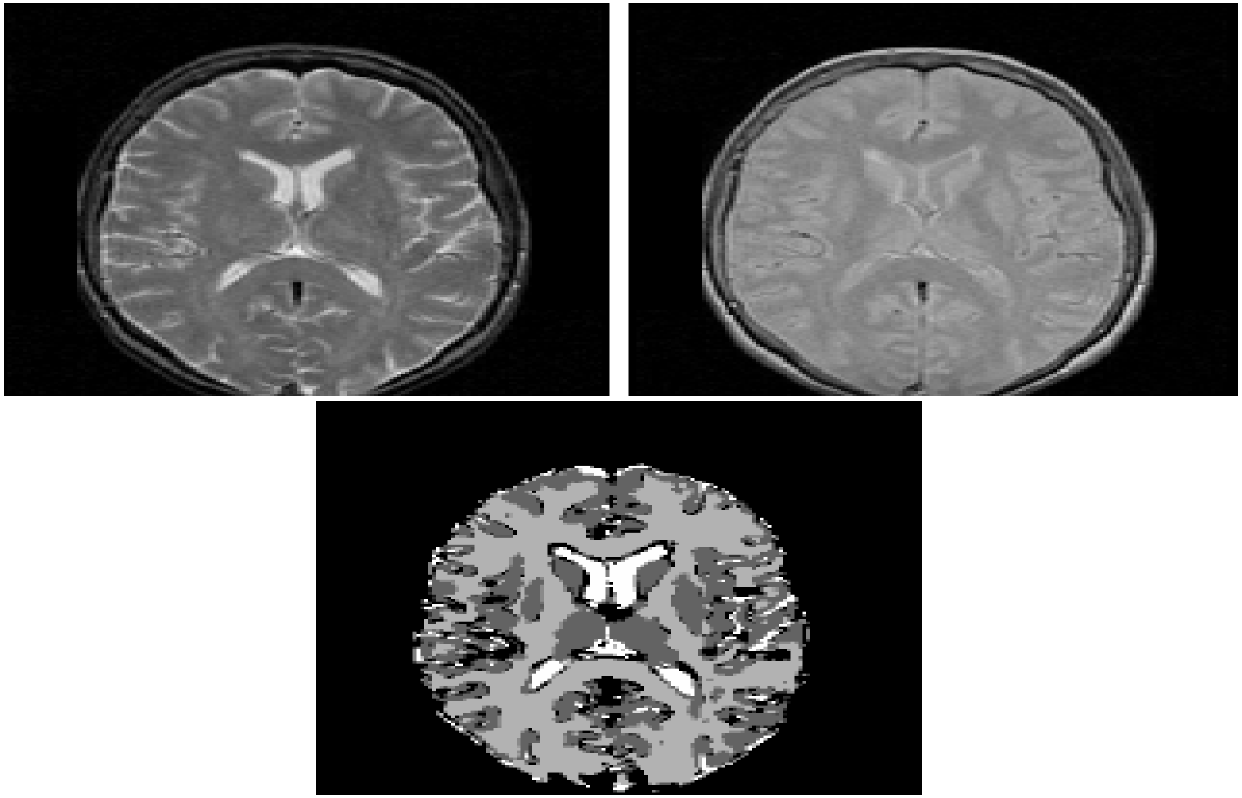
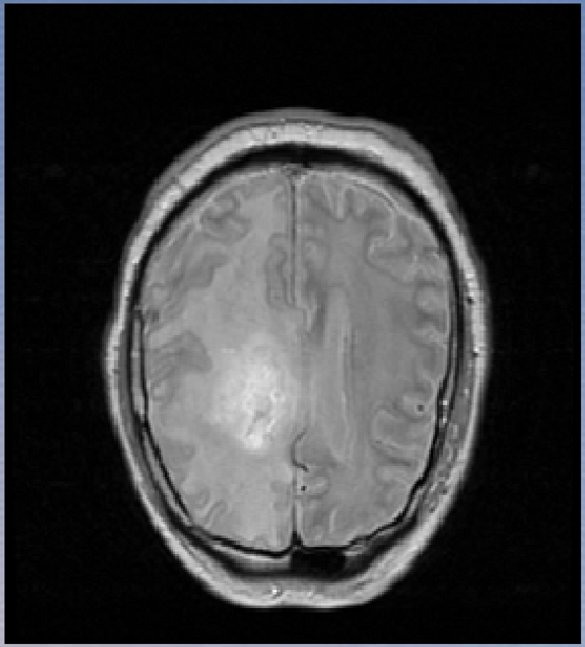
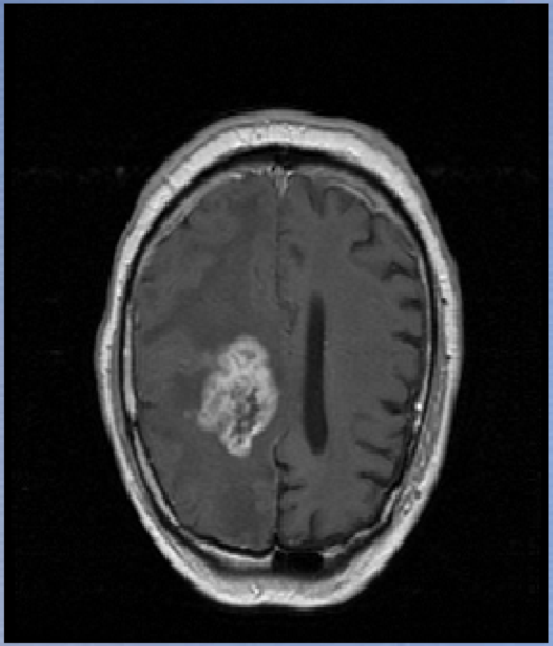


Figure 10: Segmentation of one scan with PD and T_2 images.

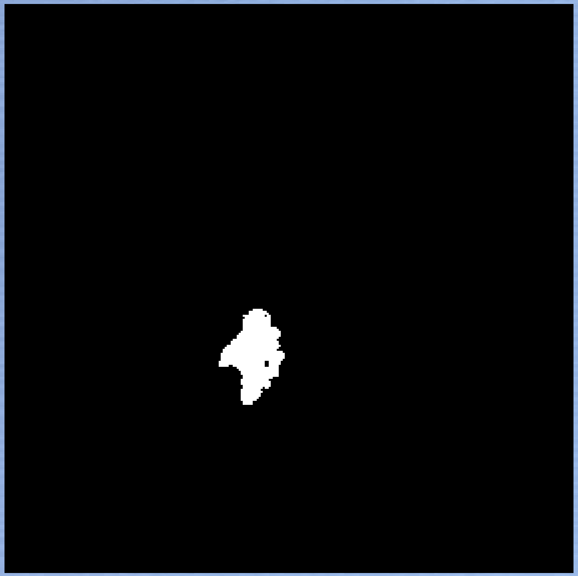
Extracting Abnormal Objects from MRI Images of the Brain





Separating Healthy Tissue from Tumor

Simulating and Planning Gamma Therapy & Surgery



Extracting Tumors from MRI
T1 and T2 Images

Imperial College London

2) RNN based Adaptive Video Compression: Combining Motion Detection and RNN Still Image Compression

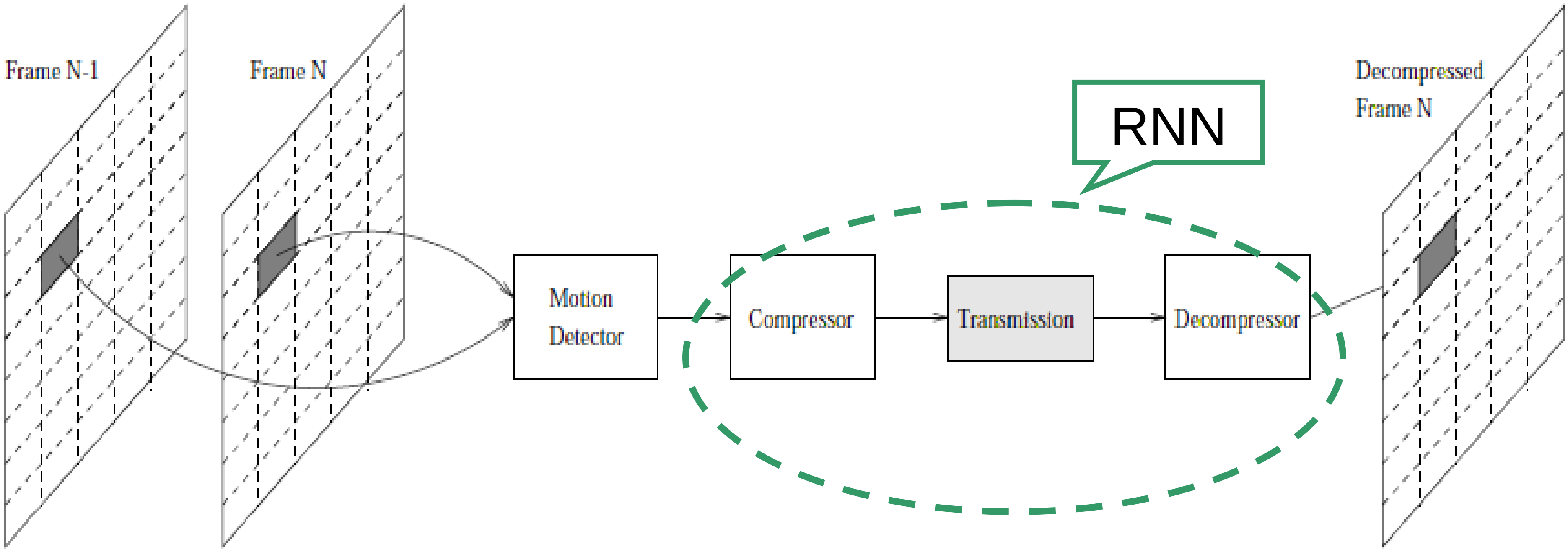
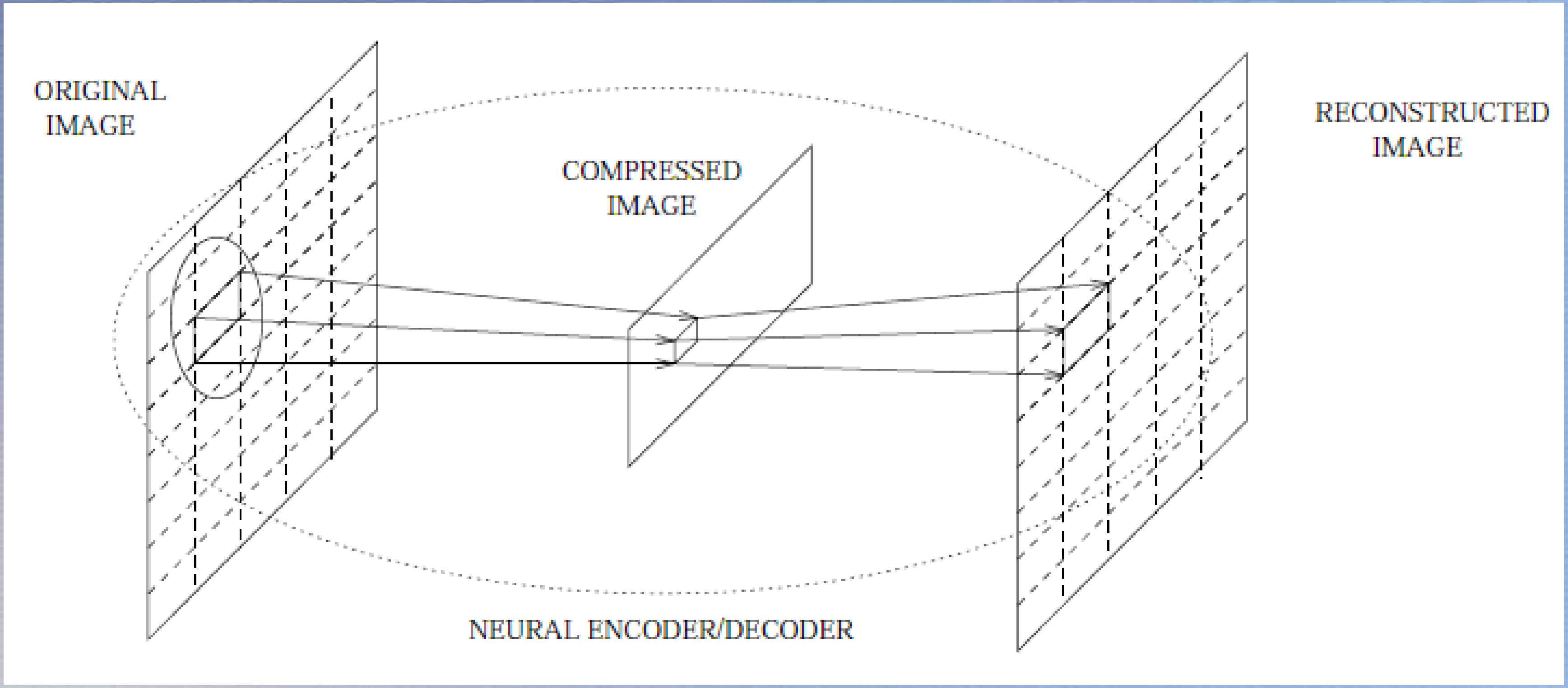


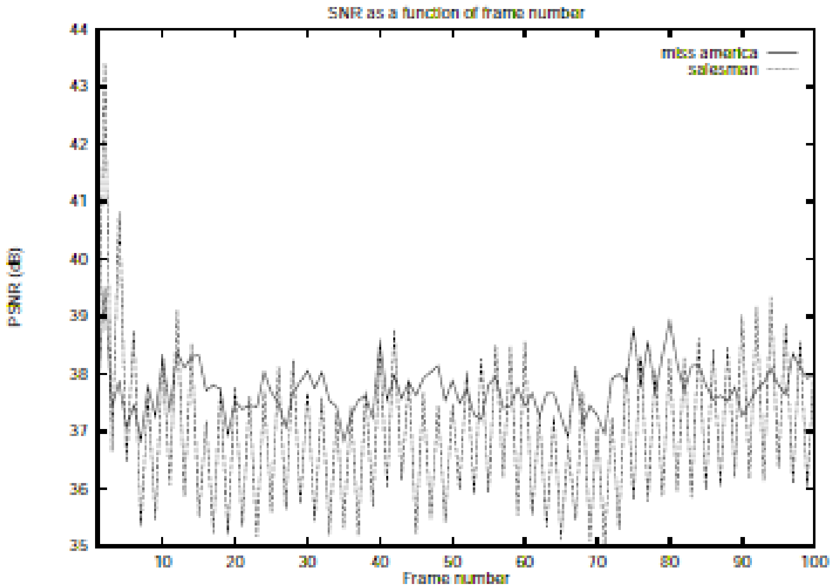
Figure 1: Block Diagram of a Video compression scheme.

Imperial College London

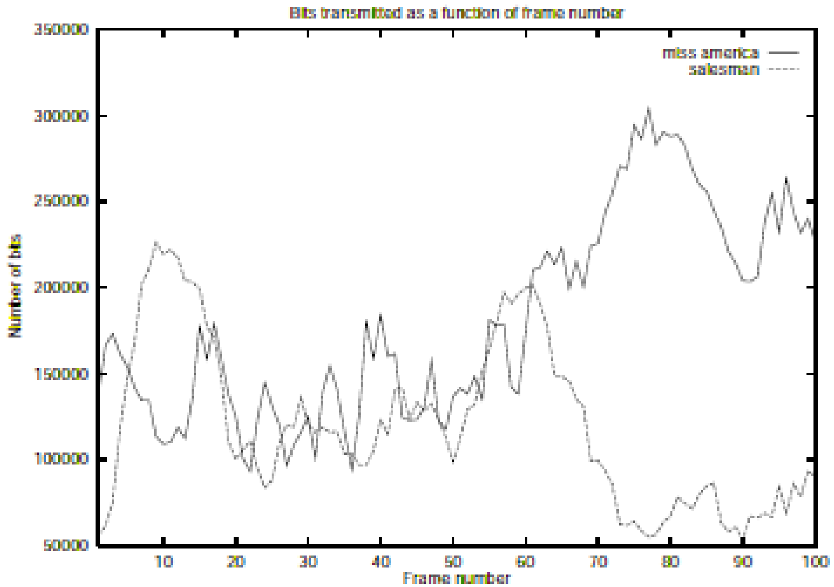
Neural Still Image Compression
Find RNN R that Minimizes
| | R(I) - I | |
Over a Training Set of Images { I }



RNN based Adaptive Video Compression



(a)



(b)

Figure 6: Experimental results for motion detection with $d = 1$: a) PSNR as a function of frame number, b) Number of bits transmitted as a function of frame number

Imperial College London

Original

After decom-
pression

Figure 30: Results of the 100th frame of the salesman sequence for $Q = 30$ and $d = 1$ (31.50 dB), 3 (30.99 dB), 5 (30.17 dB) and 7 (29.61 dB)

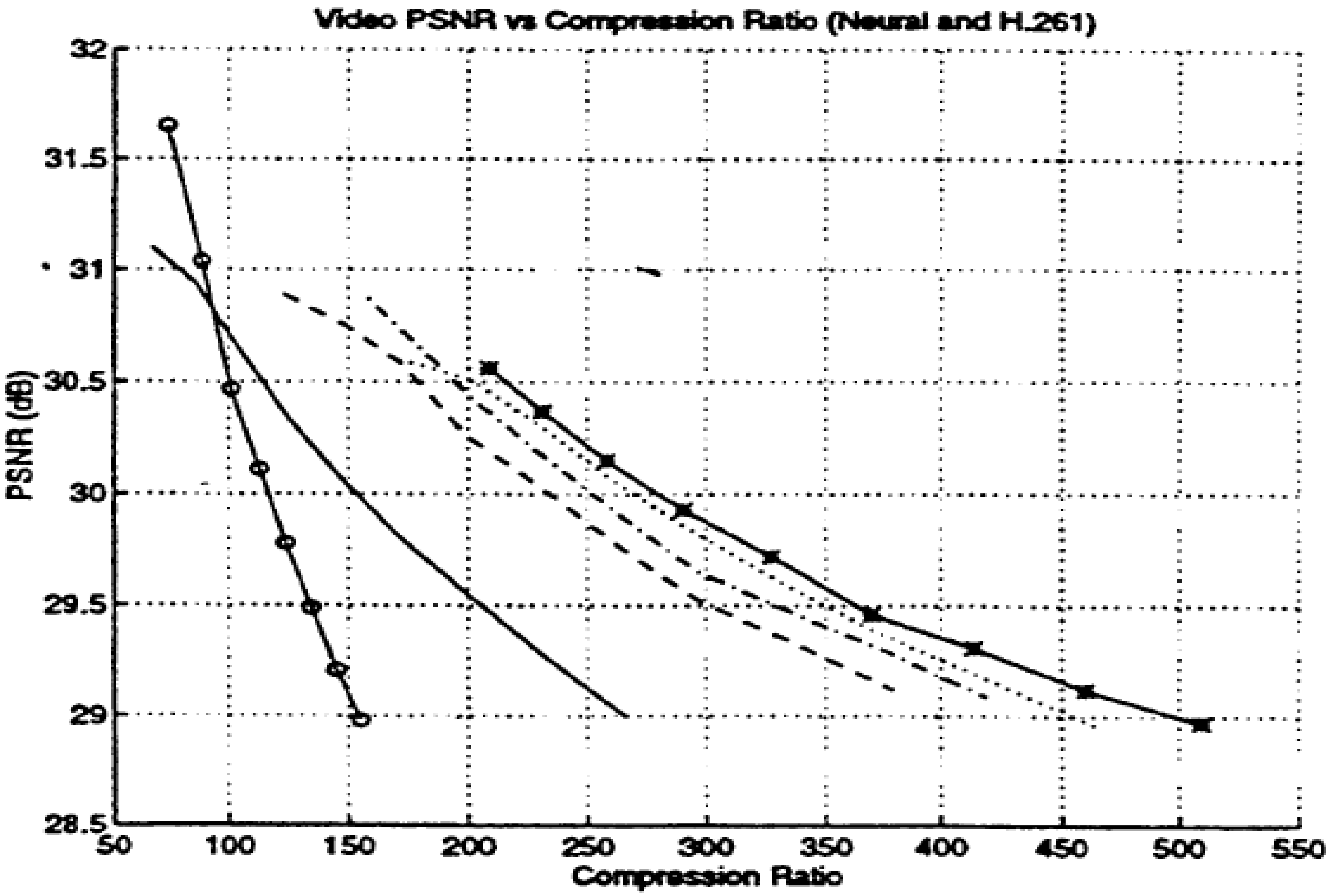


Fig. 17. Graph of video quality versus compression ratios for subsampling of one (solid), three (dashed), four (dot-dashed), five (dotted), six (solid with stars), and H.261 (solid with circles).

Imperial College
London

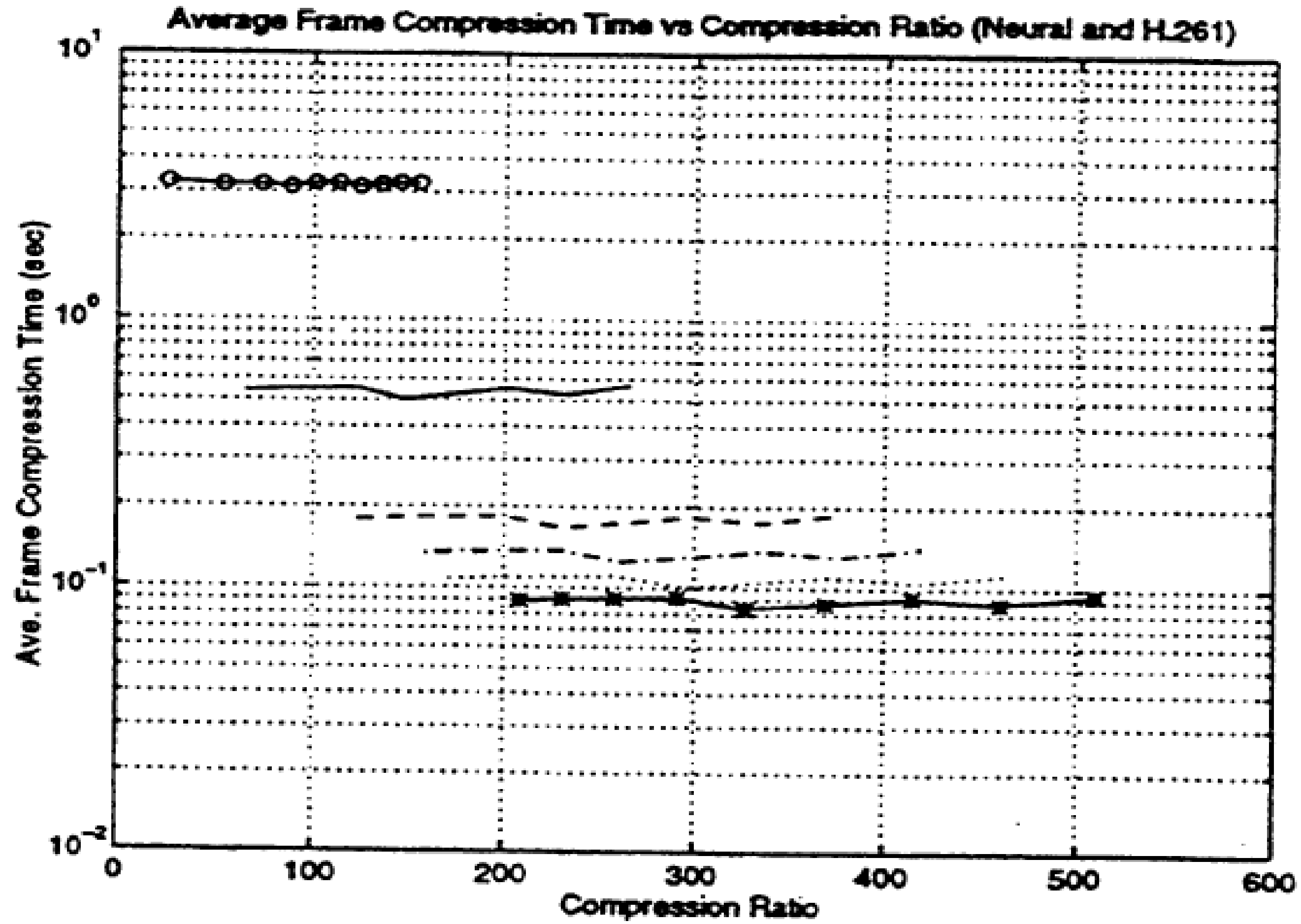


Fig. 18. Graph of compression time versus compression ratios for subsampling of one (solid), three (dashed), four (dot-dashed), five (dotted), six (solid with stars), and H.261 (solid with circles).

Imperial College London

3) Multicast Routing
Analytical Annealing with the RNN

similar improvements were obtained for (a) the Vertex Covering Problem
(b) the Traveling Salesman Problem

- Finding an optimal "many-to-many communications path" in a network is equivalent to finding a Minimal Steiner Tree. This is an NP-Complete problem
- The best purely combinatorial heuristics are the Average Distance

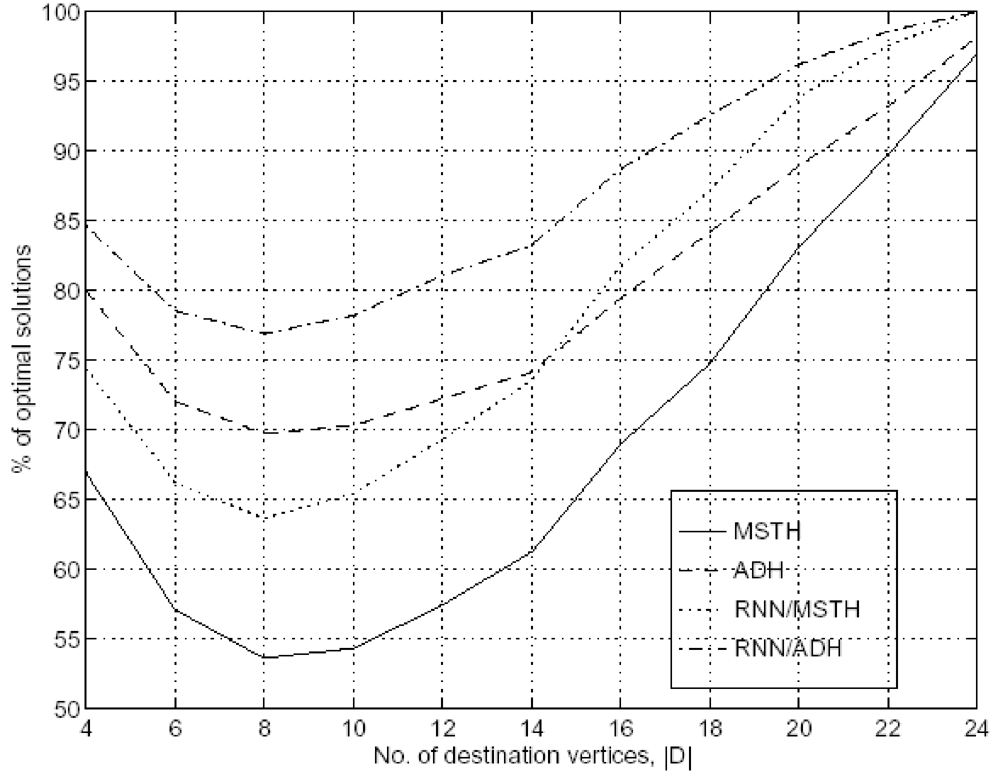


Figure 7: Percentage of optimal solutions found by each method for graphs with 25 vertices

Imperial College London

4) Learning and Reproduction of Colour Textures

- The Multiclass RNN is used to Learn Existing
- The same RNN is then used as a Relaxation Machine to Generate the Textures
- The "use" of this approach is to store textures in a highly compressed manner
- Gelenbe & Khaled, IEEE Trans. On Neural Networks (2002).

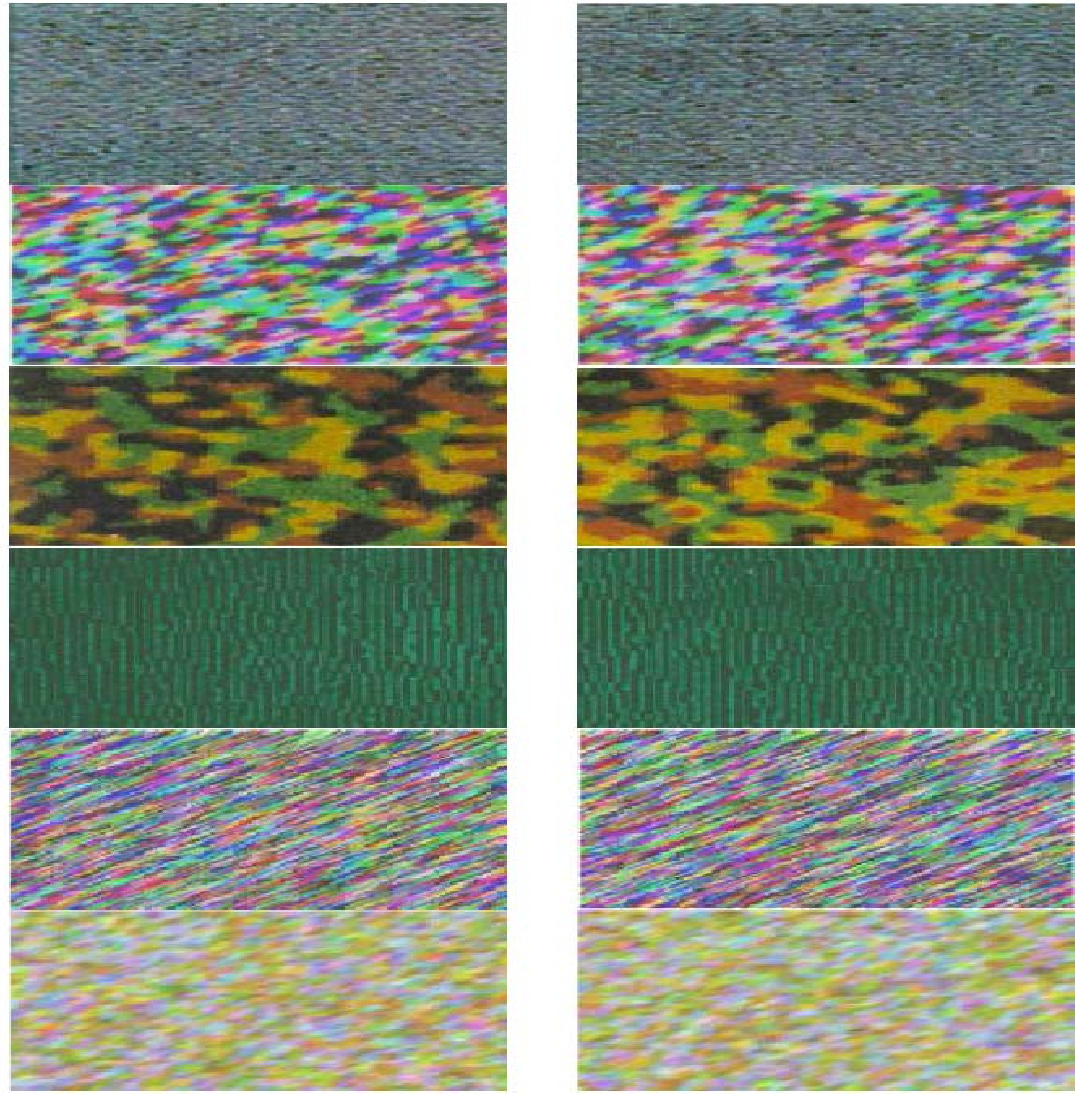


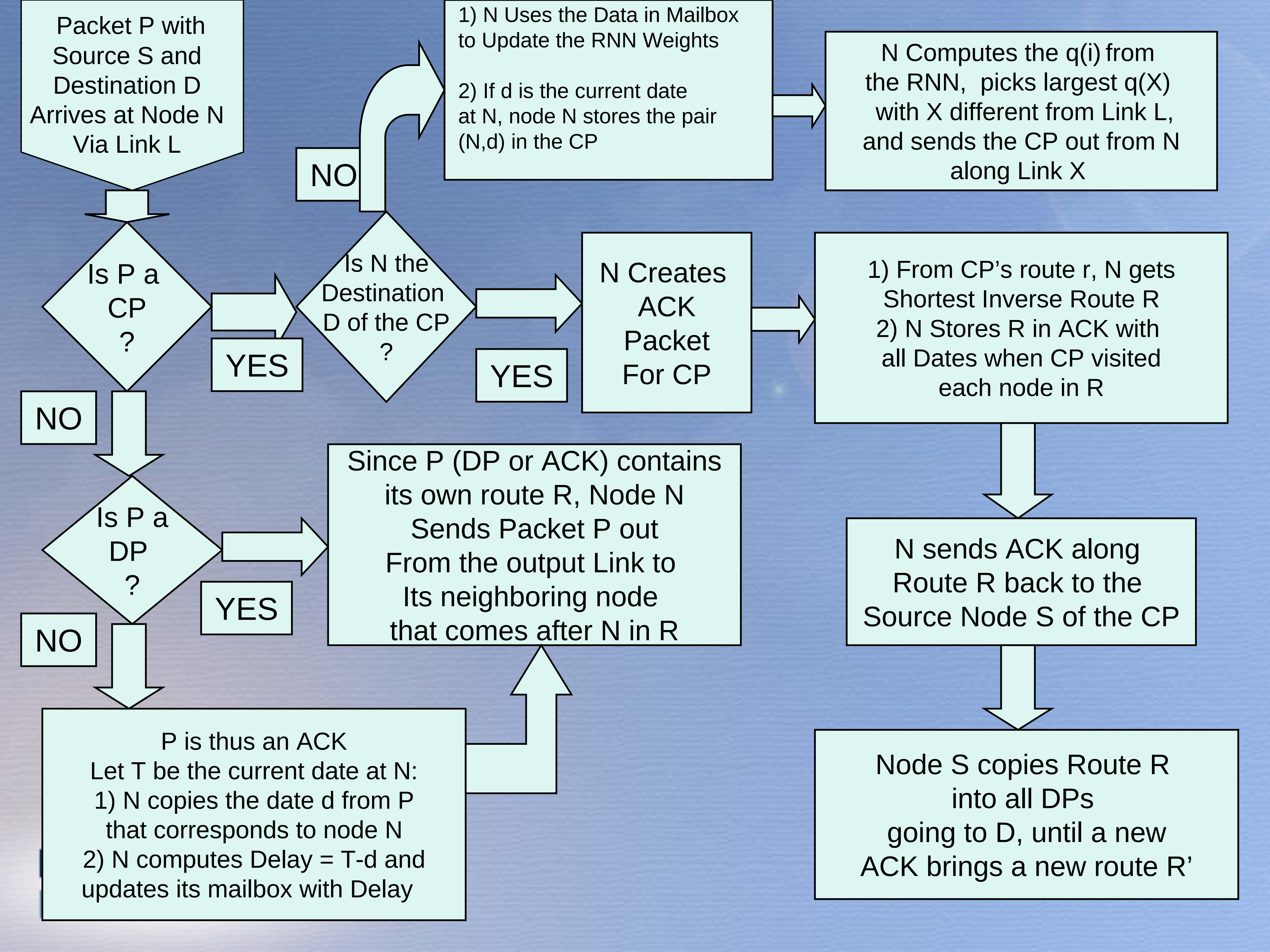
Figure 3: Comparison of synthetic color textures generated with prespecified parameters by the MCRNN (left), and learned by the MCRNN from the left-hand images and then generated by the MCRNN with the learned parameters (right)

Imperial College London

Cognitive Adaptive Routing

- Conventional QoS Goals are extrapolated from Paths, Traffic, Delay & Loss Information – this is the “Sufficient Level of Information” for Self-Aware Networking
- Smart packets collect path information and dates
- ACK packets return Path, Delay & Loss Information and deposit W(K,c,n,D), L(K,c,n,D) at Node c on the return path, entering from Node n in Class K
- Smart packets use W(K,c,n,D) and L(K,c,n,D) for decision making using Reinforcement Learning

Imperial College London



Packet P with Source S and Destination D Arrives at Node N Via Link L
Is P a CP ?
YES
NO
Is N the Destination D of the CP ?
NO
1) N Uses the Data in Mailbox to Update the RNN Weights
2) If d is the current date at N, node N stores the pair (N,d) in the CP
N Computes the q(i) from the RNN, picks largest q(X) with X different from Link L, and sends the CP out from N along Link X
YES
N Creates ACK Packet For CP
1) From CP's route r, N gets Shortest Inverse Route R
2) N Stores R in ACK with all Dates when CP visited each node in R
N sends ACK along Route R back to the Source Node S of the CP
Node S copies Route R into all DPs going to D, until a new ACK brings a new route R'
Is P a DP ?
YES
NO
Since P (DP or ACK) contains its own route R, Node N Sends Packet P out From the output Link to Its neighboring node that comes after N in R
P is thus an ACK
Let T be the current date at N:
1) N copies the date d from P that corresponds to node N
2) N computes Delay = T-d and updates its mailbox with Delay

Goal Based Reinforcement Learning in CPN

- The Goal Function to be minimized is selected by the user, e.g.

  G = [1-L]W + L[T+W]

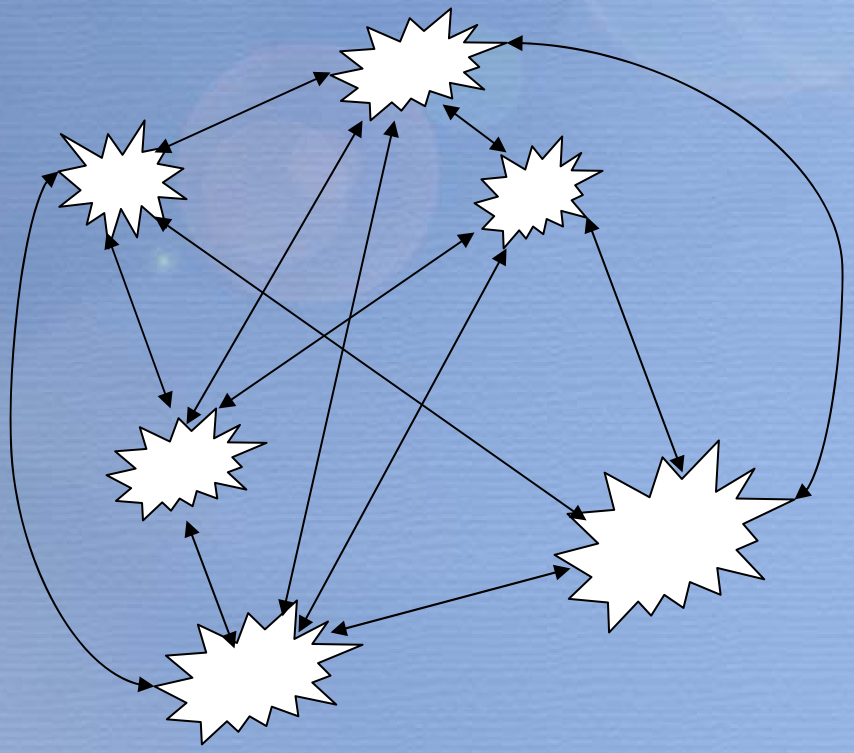
- On-line measurements and probing are used to measure L and W, and this information is brought back to the decision points
-
- The value of G is estimated at each decision node and used to compute the estimated reward R = 1/G

- The RNN weights are updated using R stores G(u,v) indirectly in the RNN which makes a myopic (one step) decision

Imperial College London

Routing with Reinforcement Learning using the RNN

- Each "neuron" corresponds to the choice of an output link in the node
- Fully Recurrent Random Neural Network with Excitatory and Inhibitory Weights
- Weights are updated with RL
- Existence and Uniqueness of solution is guaranteed
- Decision is made by selecting the outgoing link which corresponds to the neuron whose excitation probability is largest



Reinforcement Learning Algorithm

- **The decision threshold is the Most Recent Historical Value of the Reward**

$$T_l = aT_{l-1} + (1-a)R_l, R = G^{-1}$$

- **Recent Reward R_l**

If $T_{l-1} \leq R_l$ **then**

$$w^+(i, j) \leftarrow w^+(i, j) + R_l$$

$$w^-(i, k) \leftarrow w^-(i, k) + \frac{R_l}{n-2}, k \neq j$$

else

$$w^+(i, k) \leftarrow w^+(i, k) + \frac{R_l}{n-2}, k \neq j$$

$$w^-(i, j) \leftarrow w^-(i, j) + R_l$$

- **Re-normalise all weights**

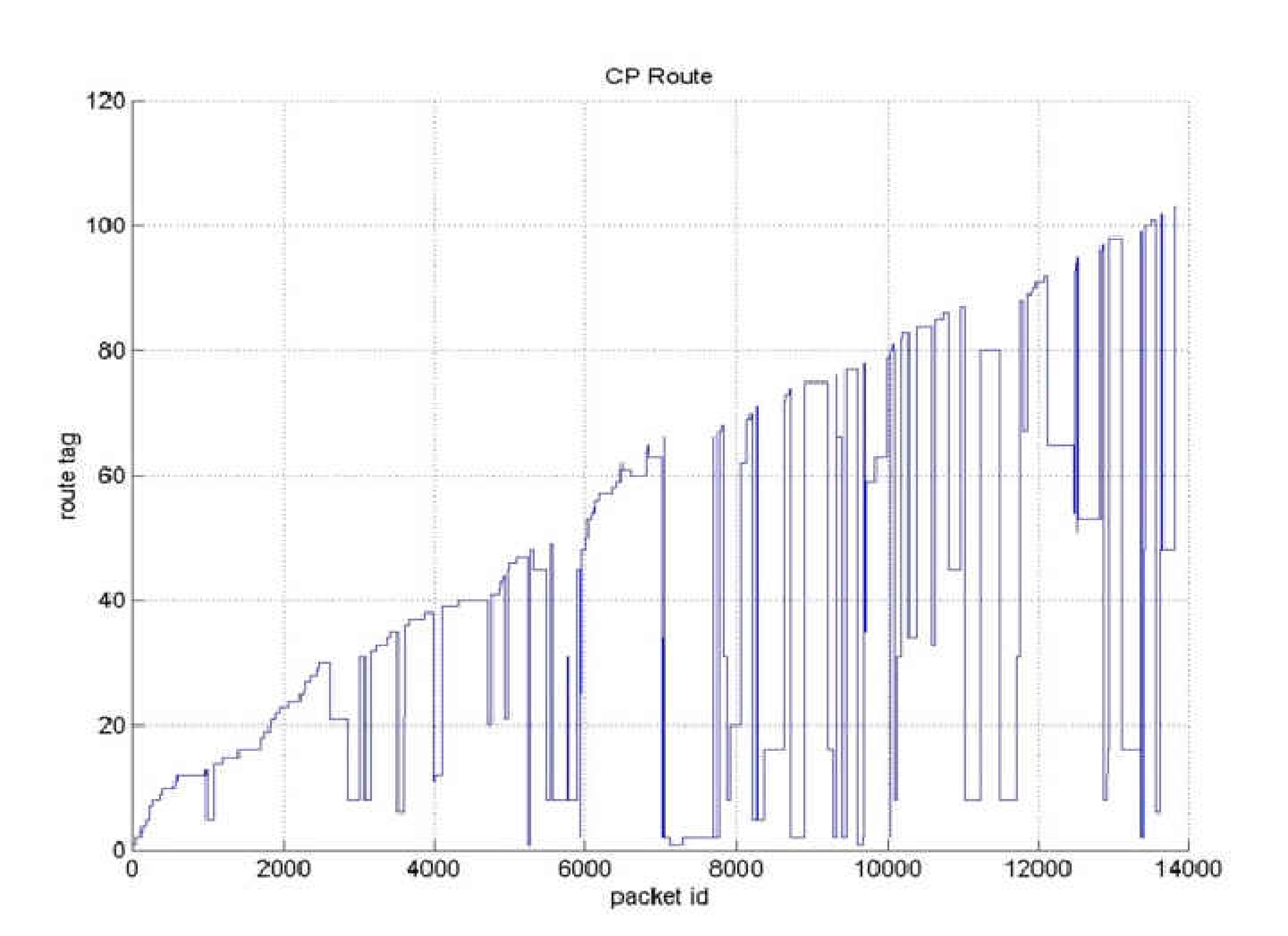
$$r_i^* = \sum_{1}^{n} [w^+(i,m) + w^-(i,m)]$$

$$w^+(i,j) \leftarrow w^+(i,j)\frac{r_i}{r_i^*}$$

$$w^-(i,j) \leftarrow w^-(i,j)\frac{r_i}{r_i^*}$$

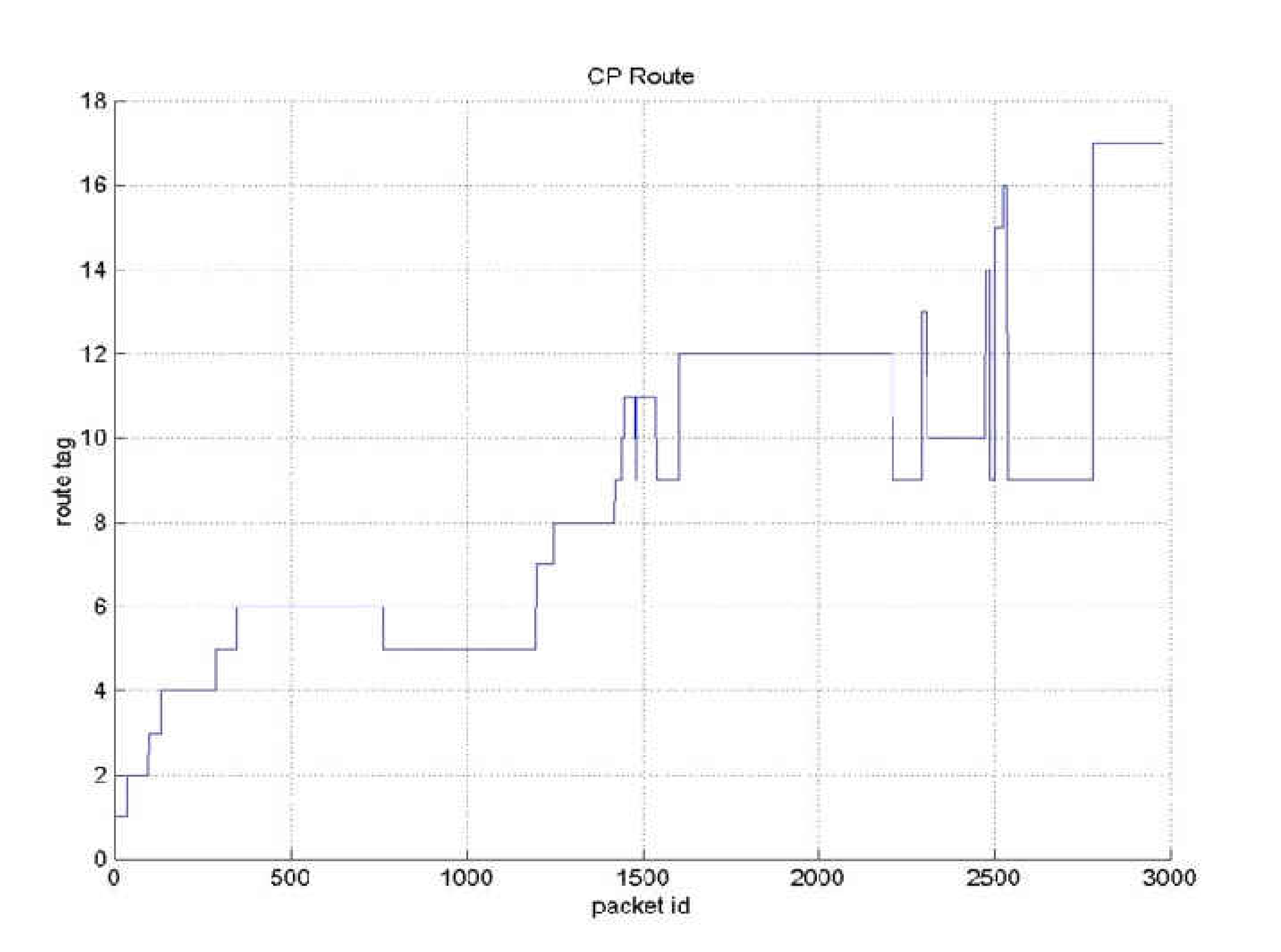
- **Compute q = (q_1, … , q_n) from the fixed-point**
- **Select Decision k such that $q_k > q_i$ for all i=1, …, n**

CPN Test-Bed Measurements
On-Line Route Discovery by Smart Packets



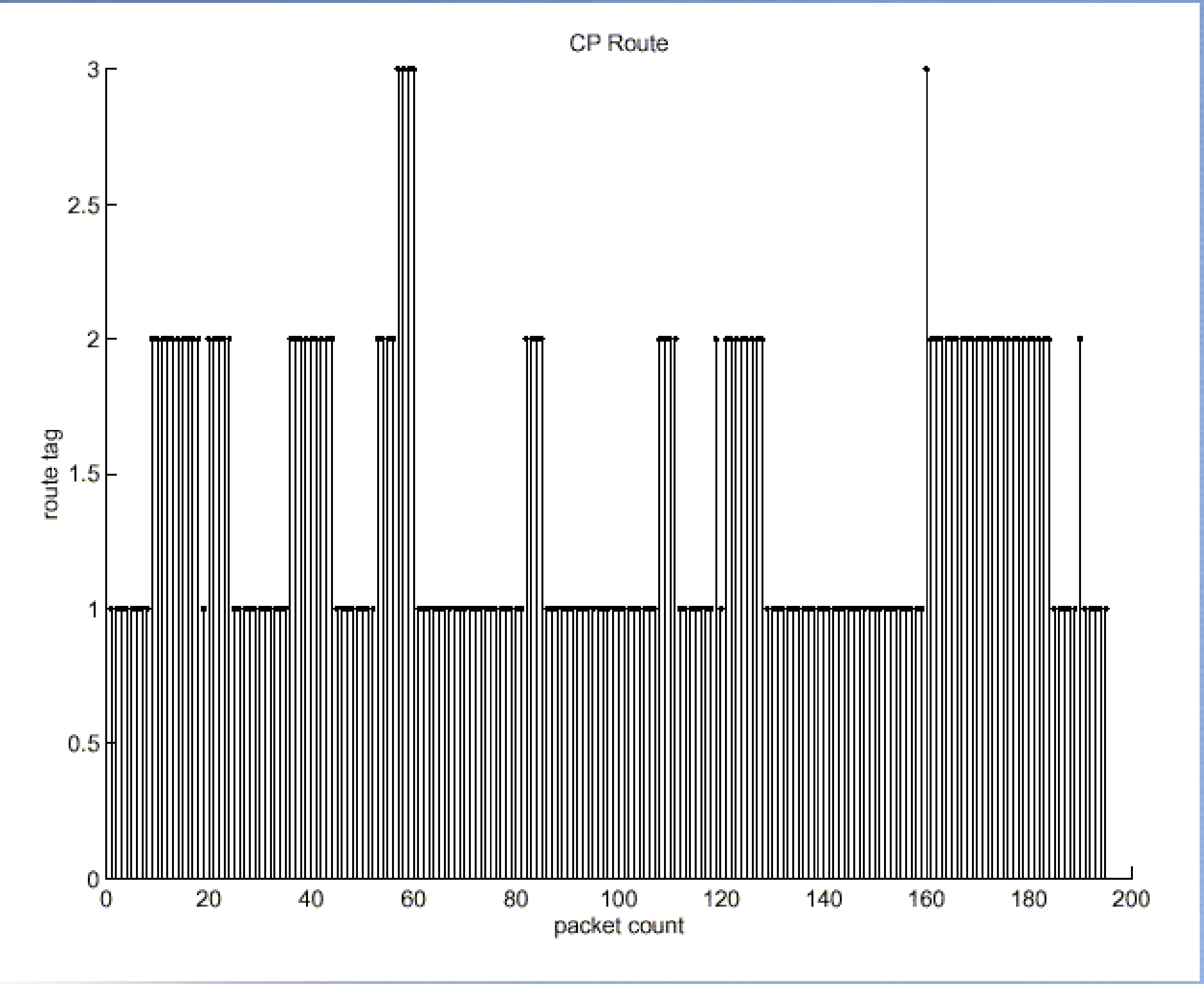
Imperial
London

CPN Test-Bed Measurements
Ongoing Route Discovery by Smart Packets



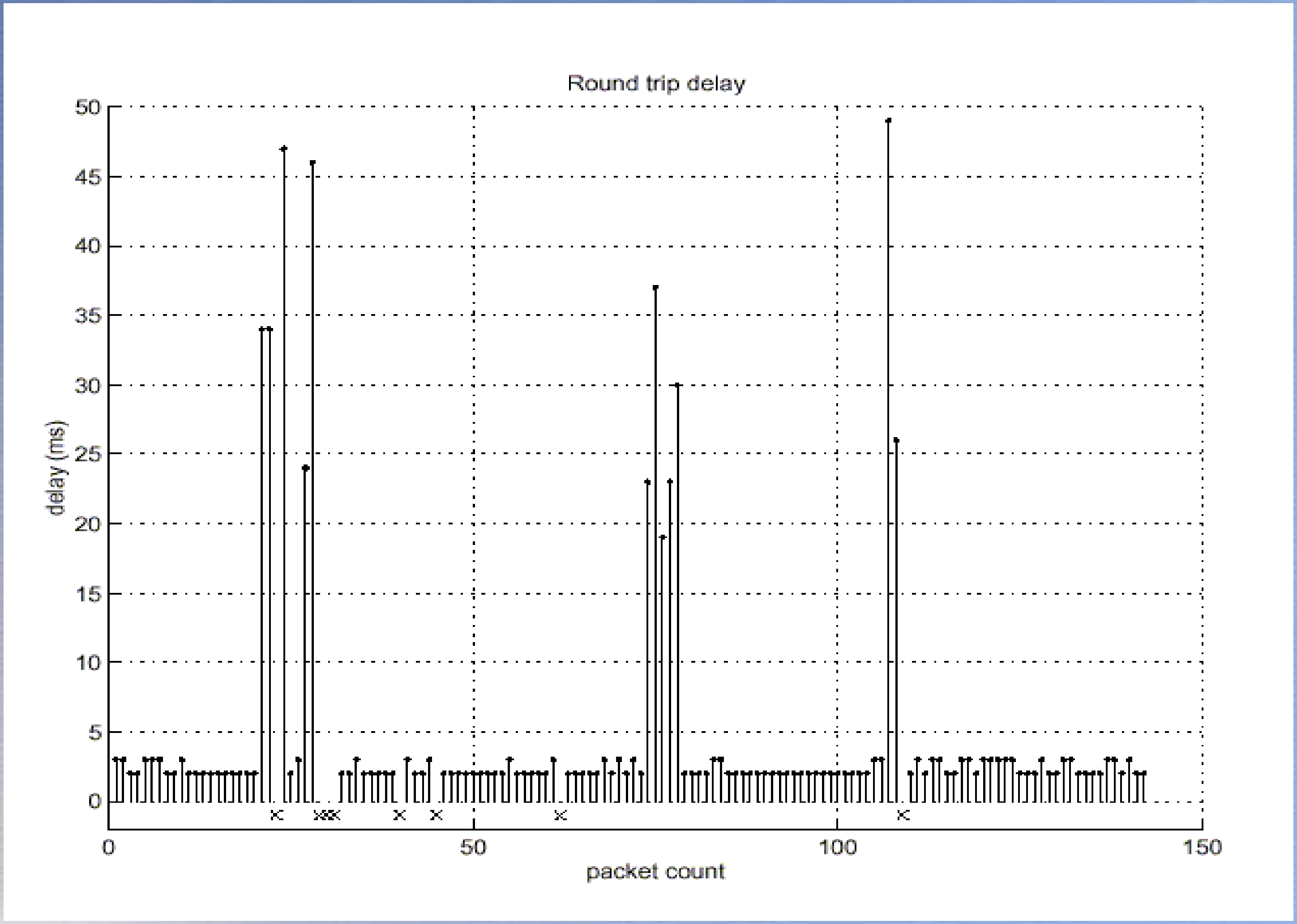
Imperial College London

Route Adaptation without Obstructing Traffic



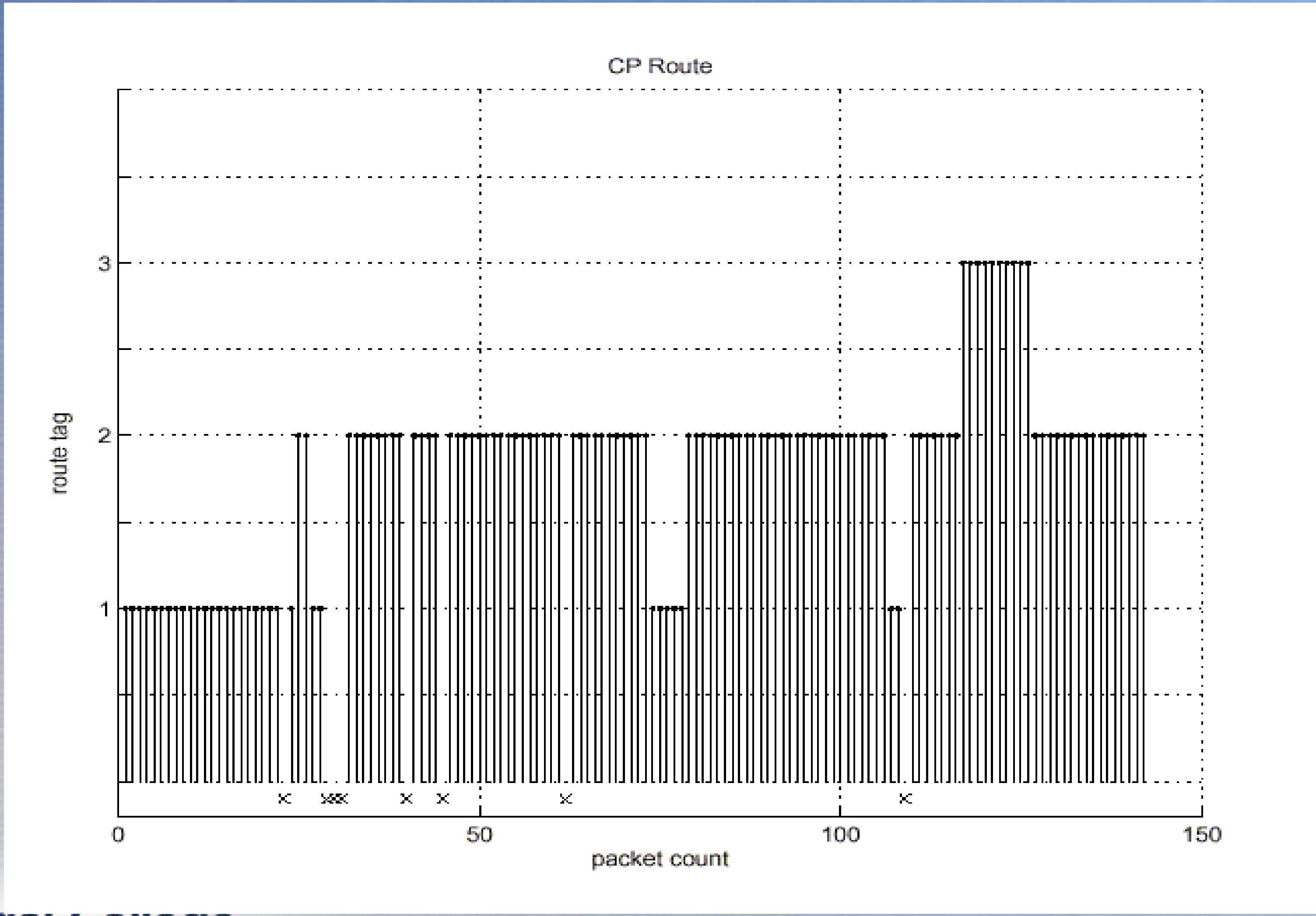
Imperial College London

Packet Round-Trip Delay with Saturating Obstructing Traffic at Count 30



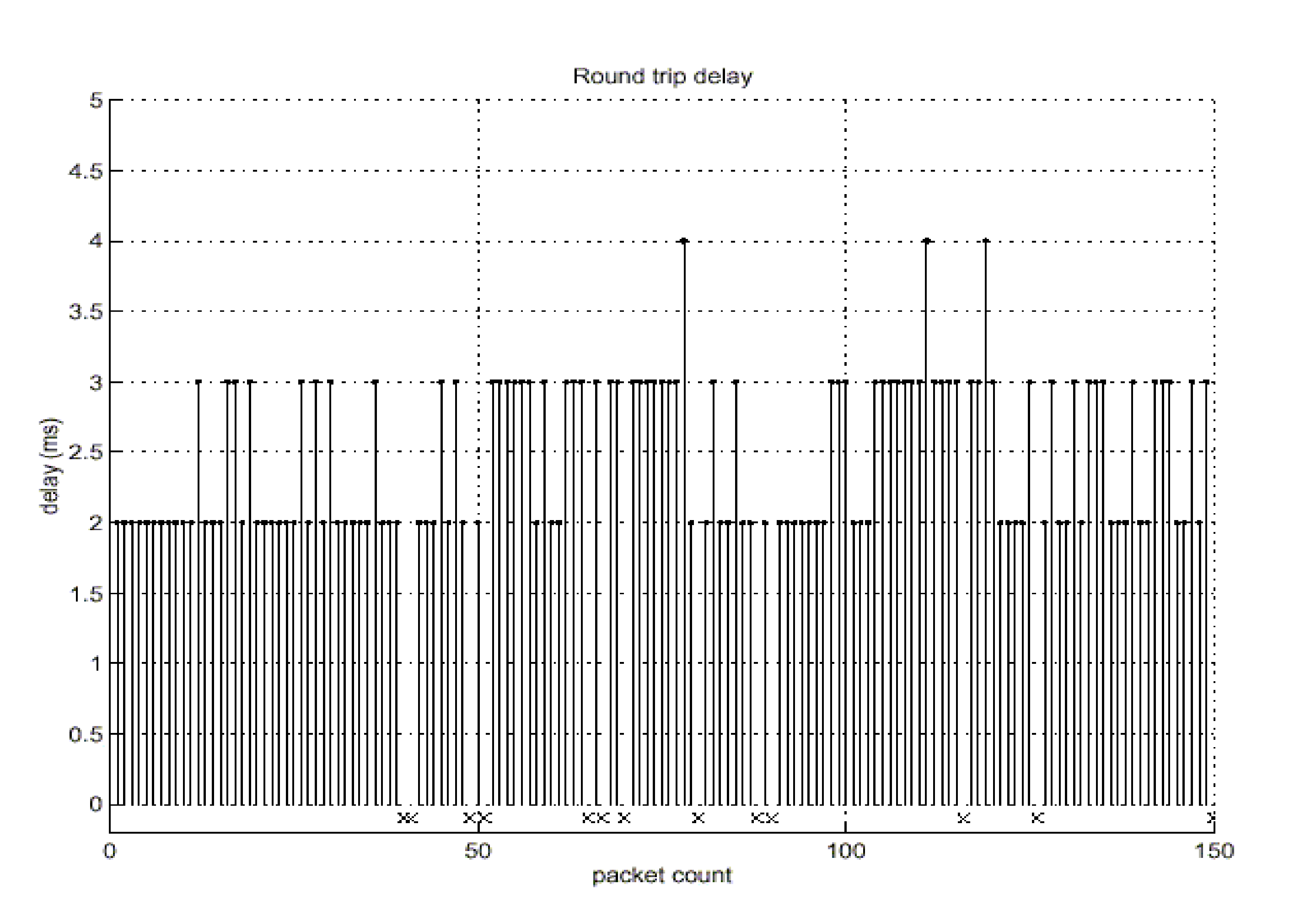
Imperial College London

Route Adaptation with Saturating Obstructing Traffic at Count 30



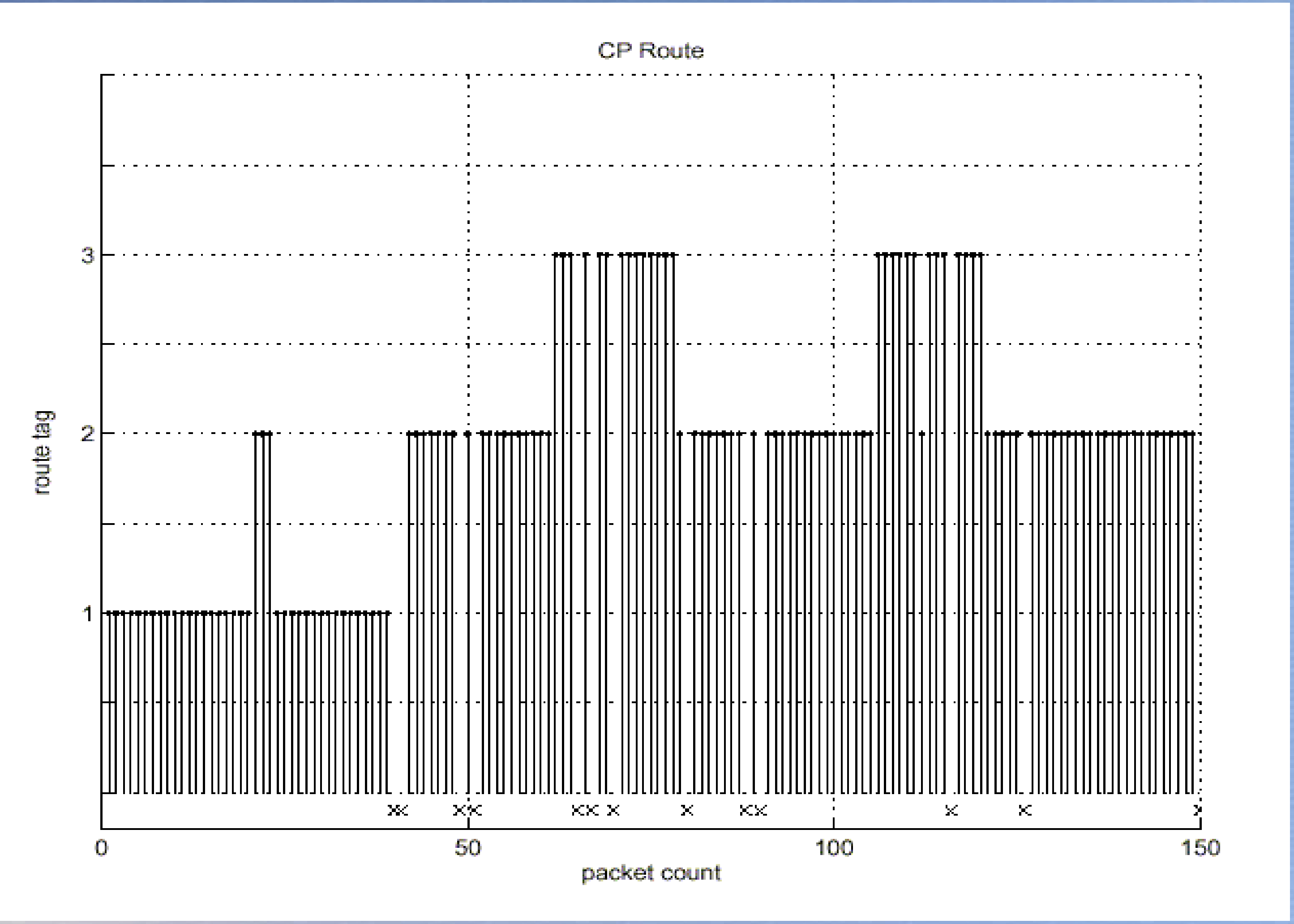
Imperial College London

Packet Round-Trip Delay with Link Failure at Count 40

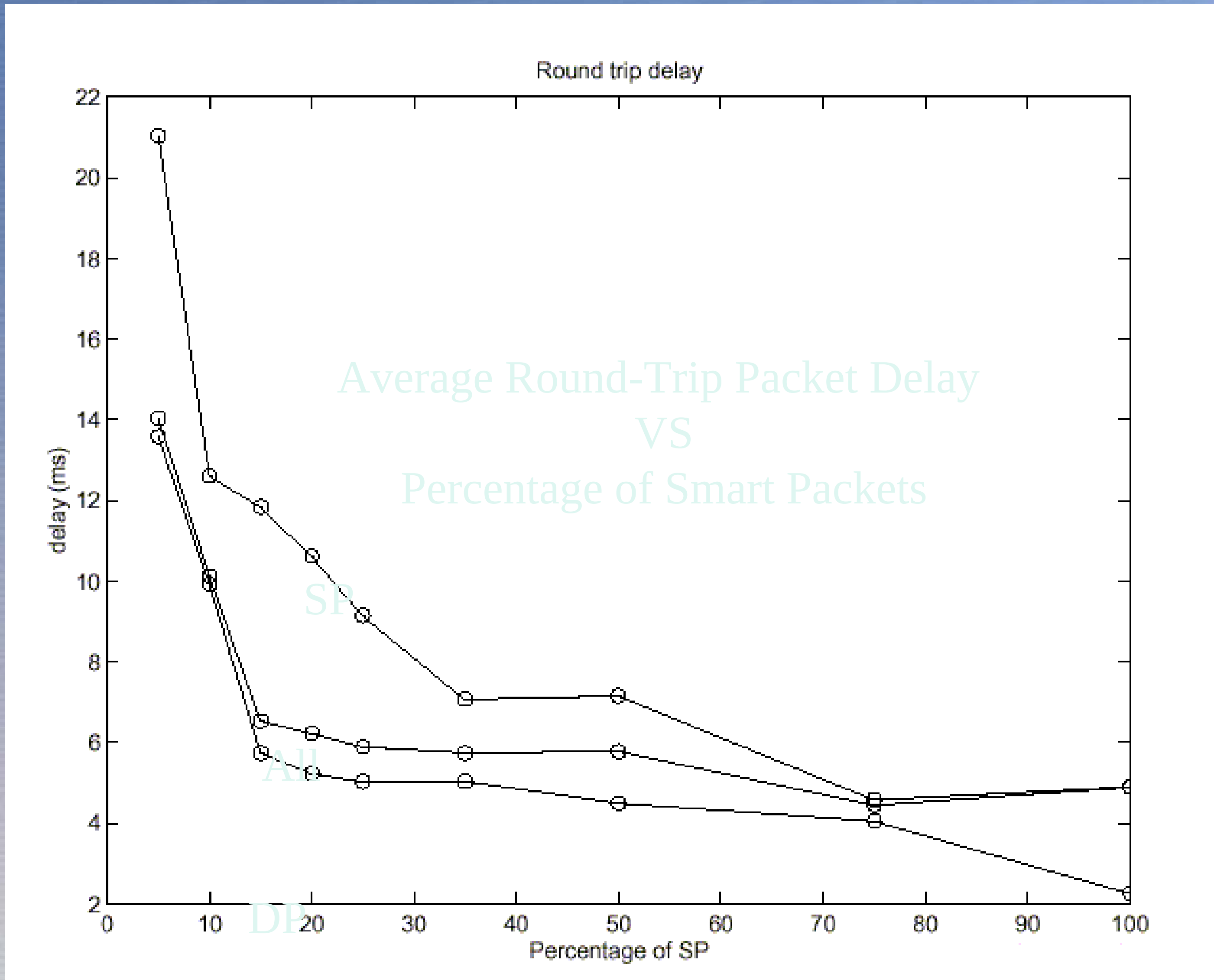


Imperial College London

Packet Round-Trip Delay with Link Failure at Count 40



Imperial College London



Imperial College
London

RNN

Other Extensions to the Mathematical Model

- Model with resets – a node can reactivate its neughbours state if they are quiescent .. Idea about sustained oscillations in neuronal networks

- Model with synchronised firing inspired by observations in vitro

- Extension of product form result and $O(n^3)$ gradient learning to networks with synchronised firing (2007)

- Hebbian and reinforcement learning algorithms

- Analytical annealing – Links to the Ising Model of Statistical Mechanics

- New ongoing chapter in queuing network theory now called “G-networks” extending the RNN

- Links with the Chemical Master Equations, Gene Regulatory Networks, Predator/Prey Population Models

Model Extensions: Synchronous Firing

$Q(i, j, m)$ is the probability that when i fires, then if j is excited it will also fire, resulting in an excitatory spike being sent to cell m. This synchronous behaviour between i and j can easily be extended to an arbitrary number of cells. Indeed, we could have a sequence of cells $j_1, \ldots, j_{n+1}, j_{n+2}$ such that $Q(j_i, j_{i+1}, j_{i+2}) = 1$ for $1 \leq i \leq n$. In this case, if cells j_1 and j_2 are excited, then eventually all the cells $j_1, \ldots, j_{n+1}, j_{n+2}$ will fire. Thus the generalised RNN model we have described can be used to model some quite general forms of synchronised firing.

2 Network behaviour in steady-state

Let the state of the network as a whole be denoted by $\underline{k}(t) = [k_1(t),\ k_2(t),\ \ldots,\ k_N(t)]$. With the assumptions that have been made about Poisson arrivals, exponential firing times, and with the given probabilities of spikes going from one cell to another, the system state is a continuous time Markov chain. As a consequence, the probability distribution of the system state $\{\underline{k}(t) : t \geq 0\}$ satisfies a set of Chapman-Kolmogorov equations. Let us use the following vectors to denote specific values of the network state, where all of these vectors' values must be non-negative:

$$\underline{k} = [k_1,\ \ldots,\ k_N]$$

$$\underline{k}_i^+ = [k_1, \ldots,\ k_i + 1,\ \ldots,\ k_N]$$

$$\underline{k}_i^- = [k_1, \ldots,\ k_i - 1,\ \ldots,\ k_N]$$

$$\underline{k}_{ij}^{+-} = [k_1, \ldots,\ k_i + 1, \ldots,\ k_j - 1,\ \ldots,\ k_N]$$

$$\underline{k}_{ij}^{++} = [k_1, \ldots,\ k_i + 1, \ldots,\ k_j + 1,\ \ldots,\ k_N]$$

$$\underline{k}_{ijm}^{++-} = [k_1, \ldots,\ k_i + 1, \ldots,\ k_j + 1, \ldots,\ k_m - 1,\ \ldots,\ k_N]$$

Imperial College London

Synchronous Firing: Solution

If the steady-state distribution $\pi(\underline{k}) = \lim_{t \to \infty} P[\underline{k}(t) = \underline{k}]$ exists, it satisfy the Chapman-Kolmogorov equations given in steady-state:

$$\pi(\underline{k}) \sum_{i=1}^{N} [\Lambda(i) + (\lambda(i) + r_i)\mathbf{1}_{\{k_i>0\}}] =$$

$$\sum_{i=1}^{N} \Big\{ \pi(\underline{k}_i^+) r_i d(i) + \pi(\underline{k}_i^-) \Lambda(i)\mathbf{1}_{\{k_i>0\}} + \pi(\underline{k}_i^+) \lambda(i)$$
$$+ \sum_{j=1}^{N} \Big[\pi\left(\underline{k}_{ij}^{+-}\right) r_i \left(p^+(i,j) + \sum_{m=1}^{N} Q(i,j,m)\right) \mathbf{1}_{\{k_j>0\}}$$
$$+ \pi\left(\underline{k}_{ij}^{++}\right) r_i p^-(i,j) + \pi(\underline{k}_i^+) r_i p^-(i,j)\mathbf{1}_{\{k_j=0\}}$$
$$+ \sum_{m=1}^{N} \pi\left(\underline{k}_{ijm}^{++-}\right) r_i Q(i,j,m)\mathbf{1}_{\{k_m>0\}} \Big] \Big\} \quad (2)$$

The following is an application of a result earlier shown in [7].

Theorem: *Let where $\lambda^-(i)$ and $\lambda^+(i)$, $i = 1, \ldots N$ be given by the following system of equations*

$$\lambda^-(i) = \lambda(i) + \sum_{j=1}^{N} r_j q_j [p^-(j,i) + \sum_{m=1}^{N} Q(j,i,m)] \quad (3)$$

$$\lambda^+(i) = \sum_{j=1}^{N} r_j q_j p^+(j,i) + \sum_{j=1}^{N}\sum_{m=1}^{N} q_j q_m r_j Q(j,m,i) + \Lambda(i) \quad (4)$$

where

$$q_i = \lambda^+(i) / \left(r_i + \lambda^-(i)\right) \quad (5)$$

If a unique non-negative solution $\{\lambda^-(i),\ \lambda^+(i)\}$ exists for the non-linear system of equations (3), (4), (5) such that $q_i < 1\ \forall i$, then:

$$\pi(\underline{k}) = \prod_{i=1}^{N} (1 - q_i)\, q_i^{k_i} \quad (6)$$

Imperial College London

An Application for the RNN-SI

Decentralised Optimisation
For Emergency Real-Time Decisions

Work Funded by
British Aerospace and the EPSRC

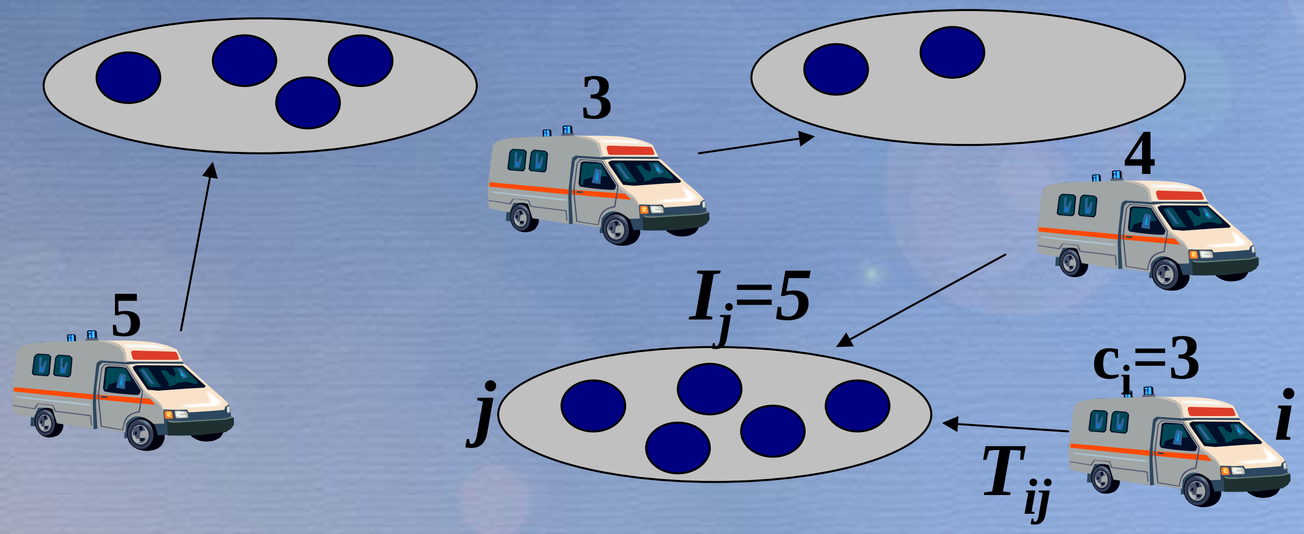
Imperial College
London

OVERVIEW

- Problem Description
- Optimisation Examples
- Solution Approaches
- Proposed Approach
- The Random Neural Network (RNN) with Synchronised Interactions
- Gradient Descent Supervised Learning for the Recurrent RNNSI
- Computational Results
- Discussion
- Future Work

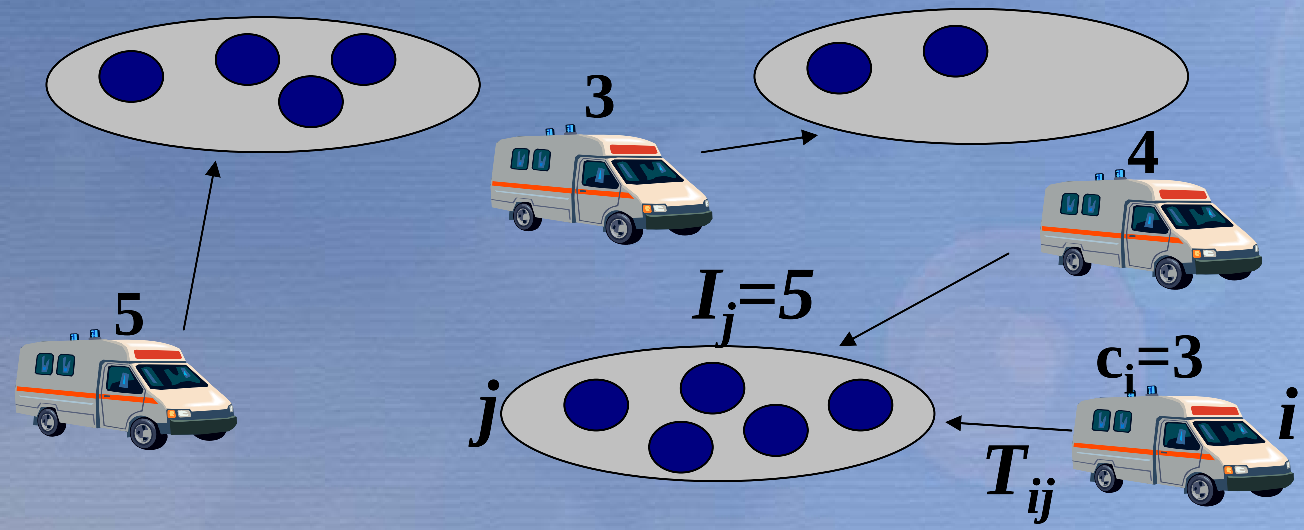
Problem Description

- Optimize a global desirable goal function under Emergency Conditions
  - Preferably no central control or distributed coordination → Distributed Decision Making
  - Real-time → Fast Decision Making
  - Complex Problems → Effective Solutions that are Close to the Optimum
  - Uncertain environments → Robustness to Variations in the Data



- ***N_L* incidents take place at given locations**
- **With *I_j* injured at incident *j***
- **The *N_U* emergency units must be dispatched to the incidents so as to optimise the response, given**
  - ***- The capacity C_i* of each emergency unit *i***
  - ***- The estimated time T_{ij}* for unit *i* to reach incident *j***

Imperial College London



The initial locations of the N_U emergency units (ambulances), and their capacities C_i are known to all of the emergency units

- When the incident occurs, the N_L incidents, their locations, and the values of I_j injured at incident j, are broadcast to all emergency units; thus estimated times T_{ij} for unit i to reach incident j become known to all
- Each of the N_U number of emergency units should then decide individually and "globally optimally" which incident they should attend to so as to optimise the response

Formulation

Find an allocation matrix $\underline{x}$ with elements x_{ij}=1 if unit *i* is allocated to incident *j* and *0* otherwise, which

minimises f($\underline{x}$) subject to a set of constraints:

$$min \ \ f(\underline{x}) = \sum_{i=1}^{N_U} \sum_{j=1}^{N_L} T_{ij} x_{ij}$$

$$s.t. \ \sum_{j=1}^{N_L} x_{ij} = 1 \ \forall i$$

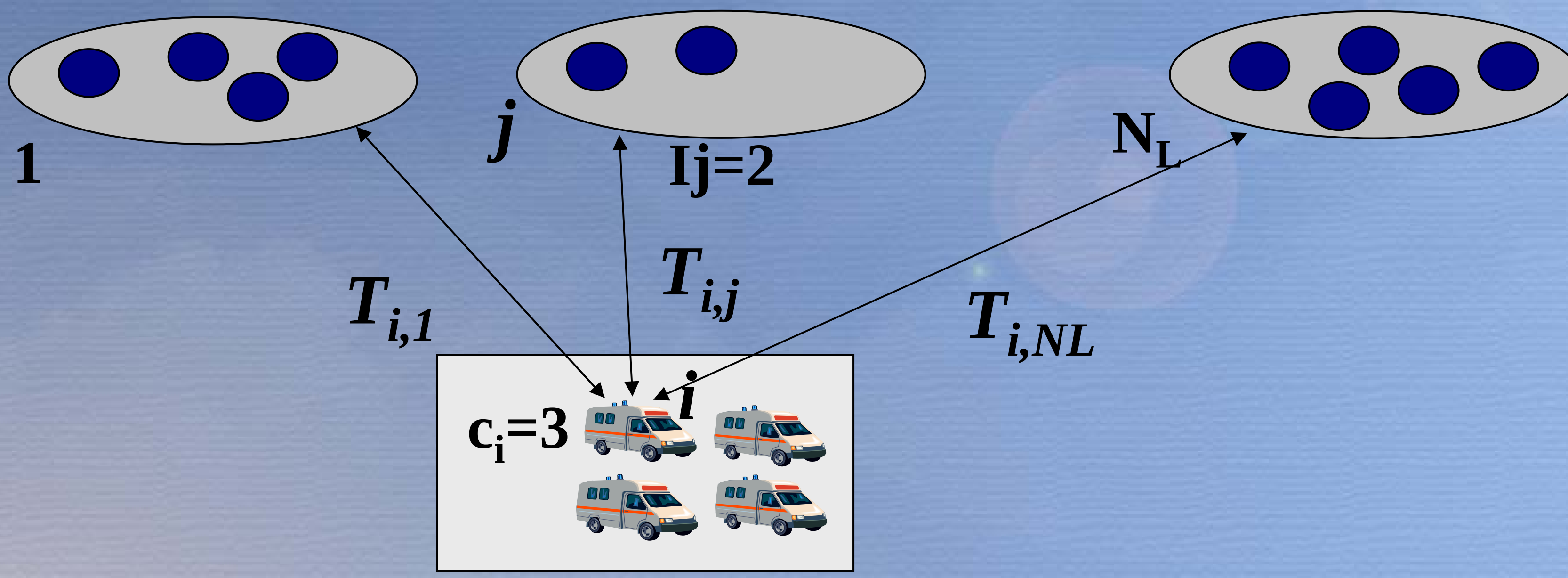
$$\sum_{i=1}^{N_U} c_i x_{ij} \geq I_j \ \forall j$$

$$x_{ij} \in \{0, 1\} \ \forall \ i, \ j$$

Imperial College London

This problem is known to be NP-Hard

How do we deal with the case that the total number of injured is higher than the total capacity of the ambulances?

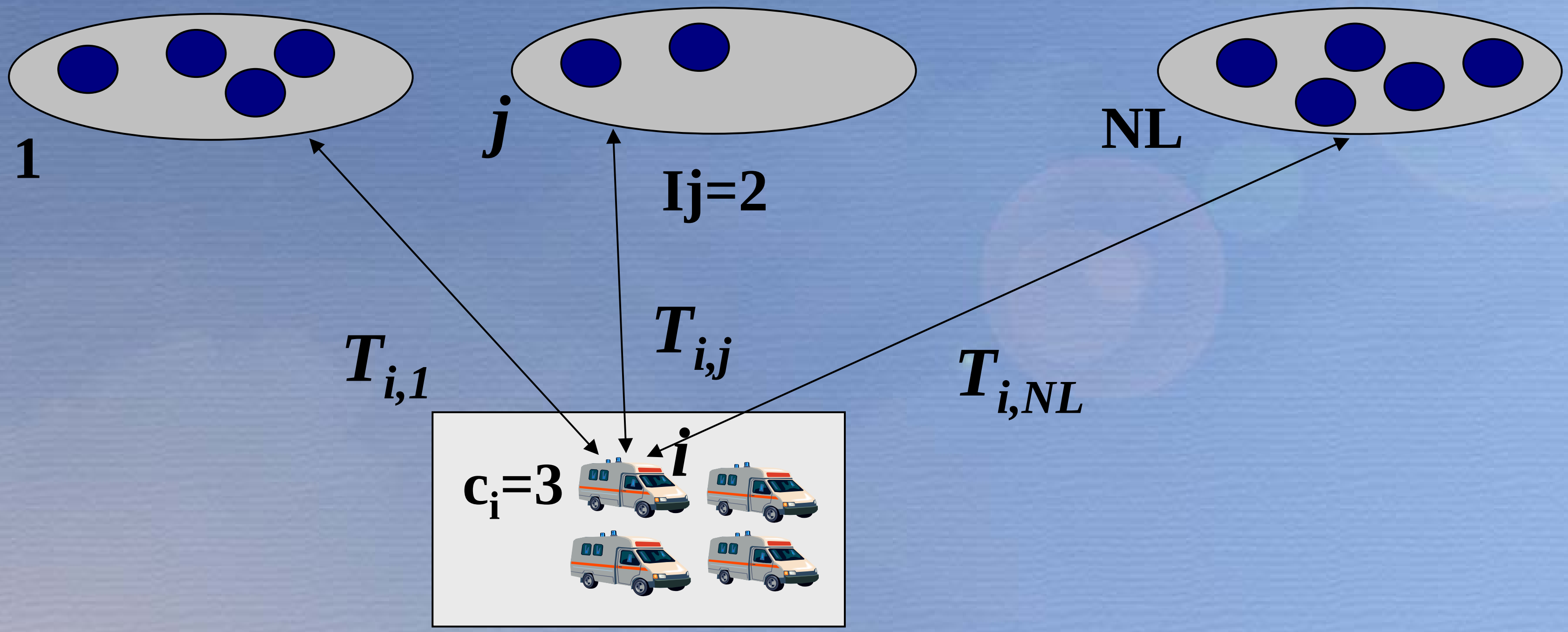


- **N_L incidents take place at given locations**
- **With I_j injured at incident *j*.**
- **The N_U emergency units must be dispatched to the incidents so as to optimise the response, given:**

  - *The capacity* C_i of each emergency unit *i*

  - *The estimated round trip time* T_{ij} for unit *i* to incident *j*

Imperial College
London



- ***Ambulances make more than one trip, until no injured are left behind***
- ***The ambulances deliver the injured to the central location***
- ***At each route k_i ambulance i goes to only one incident***

Formulation with Binary Variables

The response time for the injured collected on the k_i-th route of unit i is:

$$T^c_{ijk_i} = \sum_{l=1}^{k_i}\sum_{j=1}^{N_L} T_{ij}x_{ijl}, \quad \forall\ i,\ k_i$$

We need to find an allocation matrix x with elements $x_{ijk(i)}=1$ if unit *i* is allocated to incident *j at the* k_i route and *0* otherwise, which minimises f(x) subject to a set of constraints:

$$min\ f(x) = \sum_{i=1}^{N_U}\sum_{j=1}^{N_L}\sum_{k_i=1}^{K_i} T^c_{ijk_i} = N_L\sum_{i=1}^{N_U}\sum_{j=1}^{N_L}\sum_{k_i=1}^{K_i}\sum_{l=1}^{k_i} T_{ij}x_{ijl}$$

$$\sum_{j=1}^{N_L} x_{ijk_i} \leq 1 \quad \forall\ i,\ k_i$$

$$\sum_{i=1}^{N_U}\sum_{k_i=1}^{K_i} c_i x_{ijk_i} \geq I_j \quad \forall\ j$$

$$x_{ijk_i} \in \{0, 1\} \quad \forall\ i, j, k_i$$

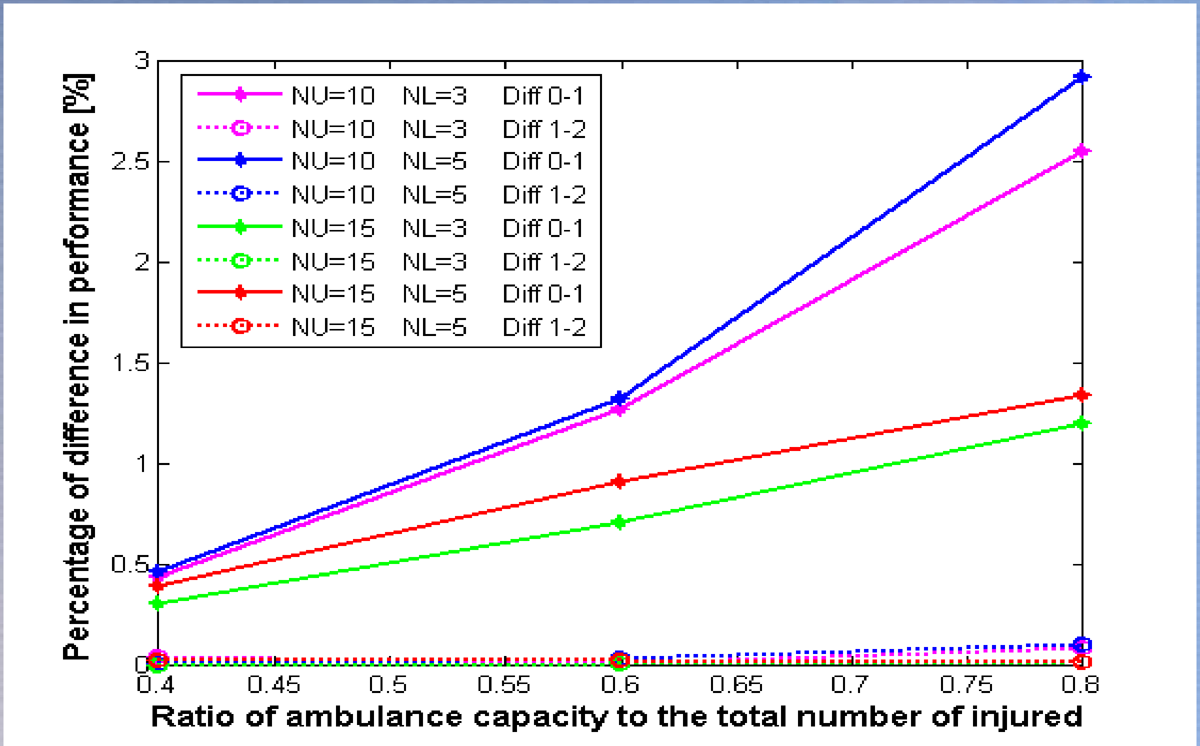
Imperial College London

How to select the K_i's

It is sufficient to assume that

$$K_i = K_{i\_rlr} + 1$$

where K_{i_rlr} *is the approximate rounded number of routes* for unit i calculated when the linear relaxation of the problem is solved



Imperial College London

Problem Formulation in Integer Variables

Given a set of allocations for unit i the optimal way to fulfill its schedule is to visit the incidents starting from the closest and finishing with the farthest

$$
\begin{aligned}
min\ f(\underline{x}) \quad = \quad & \sum_{i=1}^{N_U}\{\sum_{j=1}^{N_L} T_{ij}x_{ij}(1+x_{ij})/2 + \\
& \sum_{rank_j=1}^{N_L} T_{ij(rank_j)} \sum_{rank_m=rank_j+1}^{N_L} x_{ij(rank_j)}x_{im(rank_m)}\}
\end{aligned}
$$

$$
\sum_{i=1}^{N_U} c_i x_{ij} \geq I_j \quad \forall\ j
$$

$$
x_{ij} \epsilon \{0, 1, 2, 3, \ldots\} \quad \forall\ i, j
$$

Imperial College London

Example Problems(2)

An Example

Assume that N_L=3 and we have the following response times and allocations for unit i.

T_{i1}	T_{i2}	T_{i3}
3	2	4

X_{i1}	X_{i2}	X_{i3}
2	3	1

$T_2=T_{i2} +2T_{i2}+ 3T_{i2} =T_{i2}(1+\ldots+X_{i2})=T_{i2}X_{i2} (X_{i2}+1)/2$

$T_1=(3T_{i2}+T_{i1})+(3T_{i2}+2T_{i1})=(T_{i1}+2T_{i1})+3T_{i2}*2 = T_{i1}X_{i1} (X_{i1}+1)/2+T_{i2}X_{i2}X_{i1}$

$T_3=3T_{i2}+2T_{i1}+T_{i3}= +X_{i2}X_{i3}T_{i2}+T_{i2}X_{i2}X_{i1}+T_{i3}X_{i3} (X_{i3}+1)/2$

Imperial College London

Available Heuristic Solution Techniques

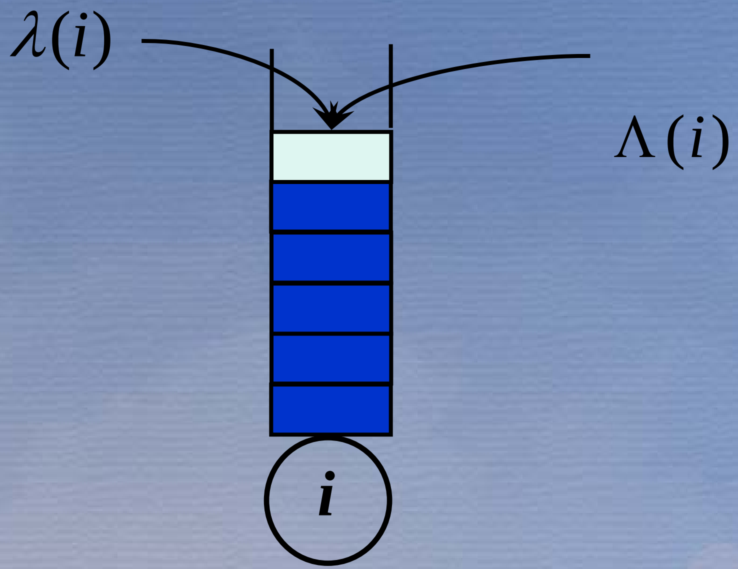
- Simulated Annealing and Other Search Techniques – long computational time, optimum can be attained
- Hopfield Network – short computational time, unlikely to find optimum
- Market/Trading Mechanisms – fast, but would require central decision element or *a priori* allocation, or coordination between agents
- Coalition formation – would require *a priori* allocations or coalitions, or coordination between agents to form coalitions
- Distributed Constraint Optimization (DCOP) – would require coordination among the agents

Imperial College London

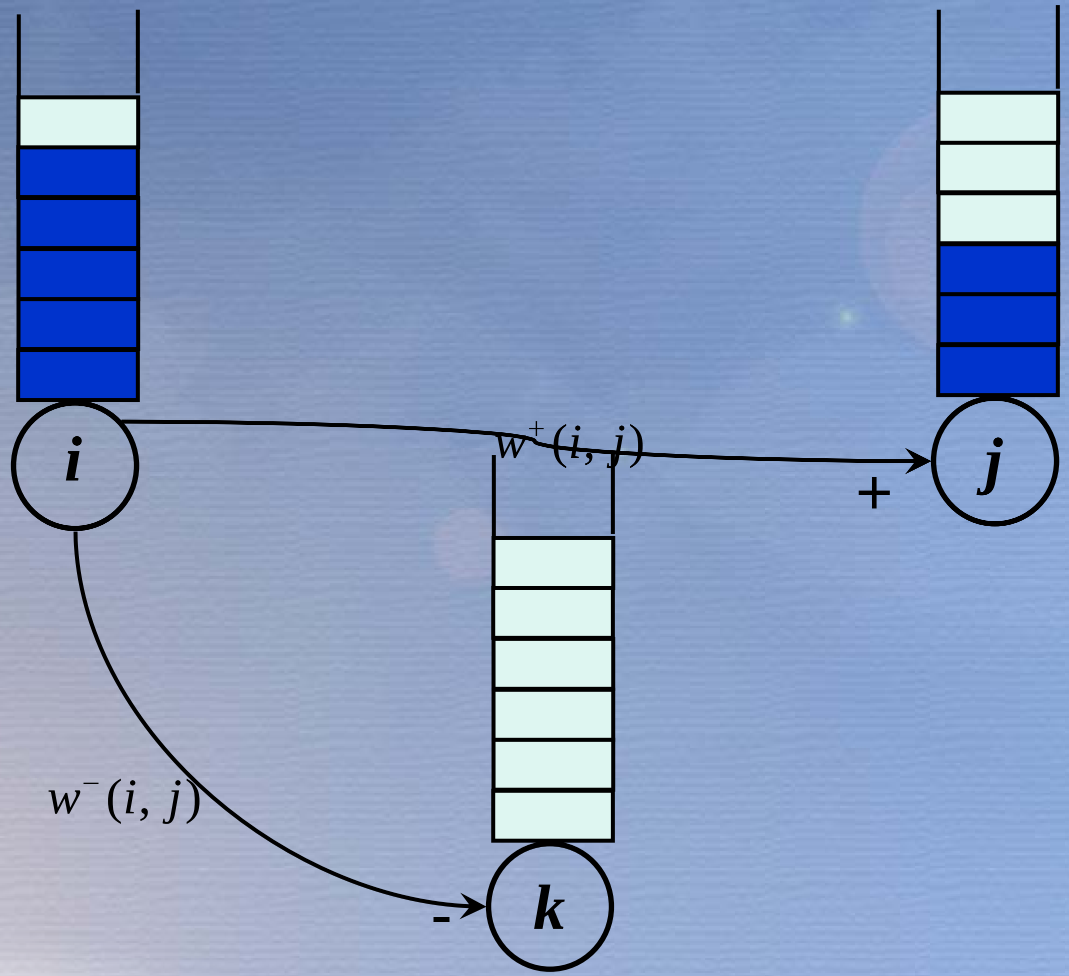
Proposed Novel Approach

- Provide a tool that acts as an “oracle” for decision making to each of the distinct and uncoordinated agents so that they take the same and non-conflicting decision if provided with the same or similar information
- The oracle is “trained” with instances of optimal decisions in the same physical context as the disaster
- Each agent uses the tool separately and receives advice as if all agents had been coordinated
- Methodology: Learning Random Neural Networks with Synchronised Interactions
- Benefits: fast and decentralised decision making, quasi optimal solutions, robust to small variations in data
- Research Novelty: the Learning Model, and the Approach to the Problem

The RNN with Synchronised Interactions

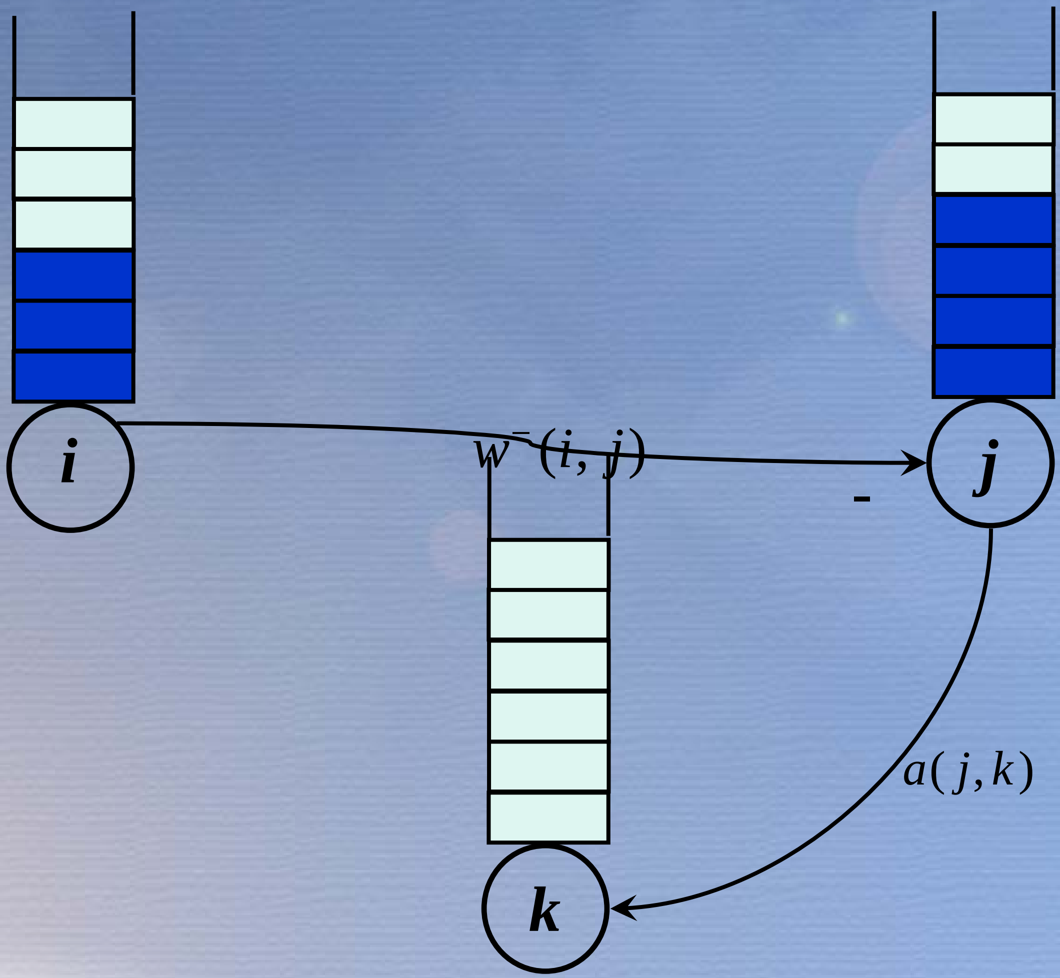


The RNN with Synchronised Interactions



Imperial College London

The RNN with Synchronised Interactions



Imperial College
London

The RNNSI: Analytical Solution

Theorem [1]: *Let where* $\lambda^-(i)$ *and* $\lambda^+(i)$, $i = 1, \ldots N$ *be given by the following system of equations*

$$\lambda^-(i) = \lambda(i) + \sum_{j=1}^{N} q_j [w^-(j,i) + \sum_{m=1}^{N} w^-(j,i) a(i,m)] \qquad (1)$$

$$\lambda^+(i) = \sum_{j=1}^{N} q_j w^+(j,i) + \sum_{j=1}^{N} \sum_{m=1}^{N} q_j q_m w^-(j,m) a(m,i) + \Lambda(i) \qquad (2)$$

where

$$q_i = \lambda^+(i) / \left(r_i + \lambda^-(i) \right) \qquad (3)$$

If a unique non-negative solution $\{\lambda^-(i),\ \lambda^+(i)\}$ *exists for the non-linear system of equations (1), (2), (3) such that* $q_i < 1\ \forall i$ *,then:*

$$\pi\left(\underline{k}\right) = \prod_{i=1}^{N} \left(1 - q_i\right) q_i^{k_i} \qquad (4)$$

According to the Theorem if $q_i<1$ for every i then in steady state solution the RNNSI is given by (4)

Gradient Learning for the RNNSI

- The network is presented with input patterns $X_k=[\underline{\Lambda}_k,\underline{\lambda}_k]$ and the corresponding desired outputs $\underline{y}_k$.
- The purpose is to find values for the weights *w(u,v)* so that the cost function *E* is minimized.

$$E = \frac{1}{2}\sum_{i=1}^{N} c_i (q_i - y_i)^2, c_i \geq 0$$

- The weights are updated after every pattern using the gradient descent rule:

$$w_{k,n+1}(u,v) = w_{k,n}(u,v) - \eta \frac{\partial E}{\partial w(u,v)} = w_{k,n}(u,v) - \eta \sum_{i=1}^{N} c_i (q_i - y_i) \frac{\partial q_i}{\partial w(u,v)} \quad (5)$$

where *n* denotes the update step, *η > 0* is the "learning rate" and the partial derivative of the cost function on the right hand side is evaluated using the *n-th* computed values of the weights

- The key step of the computation is the calculation of the $\frac{\partial q_i}{\partial w(u,v)}$

It can be shown that its has the form below and is therefore of complexity $O(n^3)$

$$\frac{\partial \underline{q}}{\partial w(u,v)} = \underline{\gamma}^s(u,v)\,(\mathbf{I} - \mathbf{W})^{-1} \quad (6)$$

Imperial College London

Gradient Descent Supervised Learning

- **Steps of the gradient descent learning algorithm**

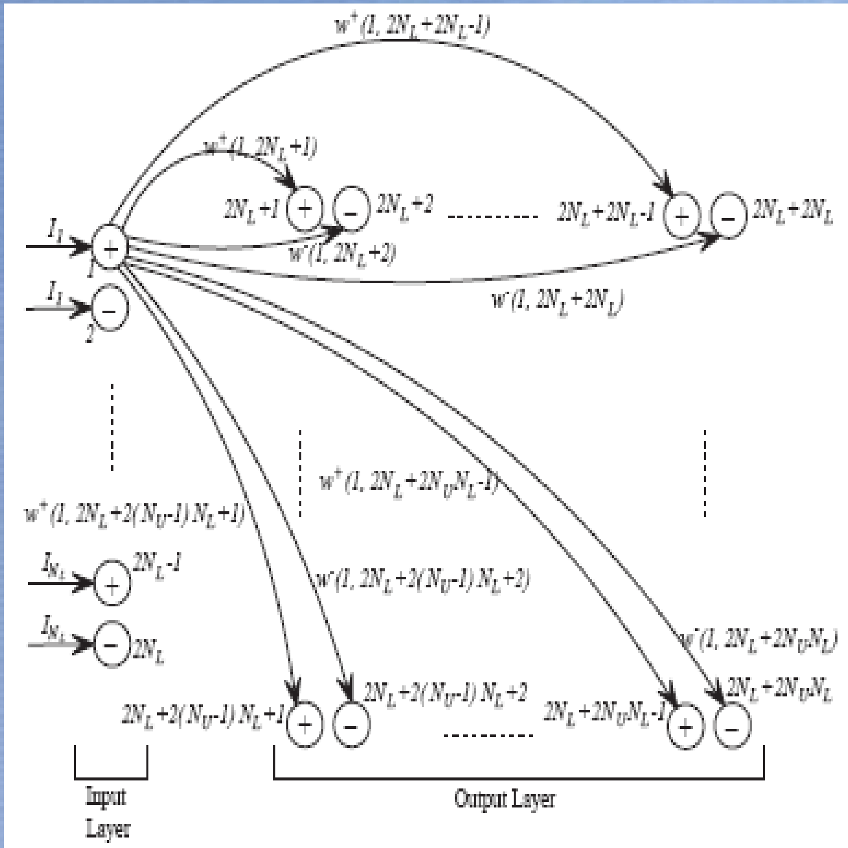
1. **Initialise the excitation-inhibition matrices** $W^+ = \{w^+(i,j)\}$, $W^- = \{w^-(i,j)\}$ **and the synchronisation matrix** $A = \{\alpha(i,j)\}$
2. **For a particular pattern** X_k **initialise the parameters** $\underline{\Lambda}_k$ **and** $\underline{\lambda}_k$ **as well as the desired output** $\underline{y}_k$
3. **Solve the system of non-linear equations (1)-(3) based on the above values**
4. **Using the values for** q_{ik} **obtained solve the 3 linear equations (6) for the weights** $w^+(u,v)$, $w^-(u,v)$ **and** $\alpha(u,v)$ **where** $\gamma^+(u,v)$, $\gamma^-(u,v)$ **and** $\gamma^\alpha(u,v)$ **are functions of known parameters**
5. **Exploiting the results from Steps 3 and 4, update the weight matrices** W^+, W^- **and** A **using the general equation (5)**
6. **Repeat steps 2,3,4,5 until convergence to a stopping criterion**

Imperial College London

Learning for the Ambulance Dispatching Problem

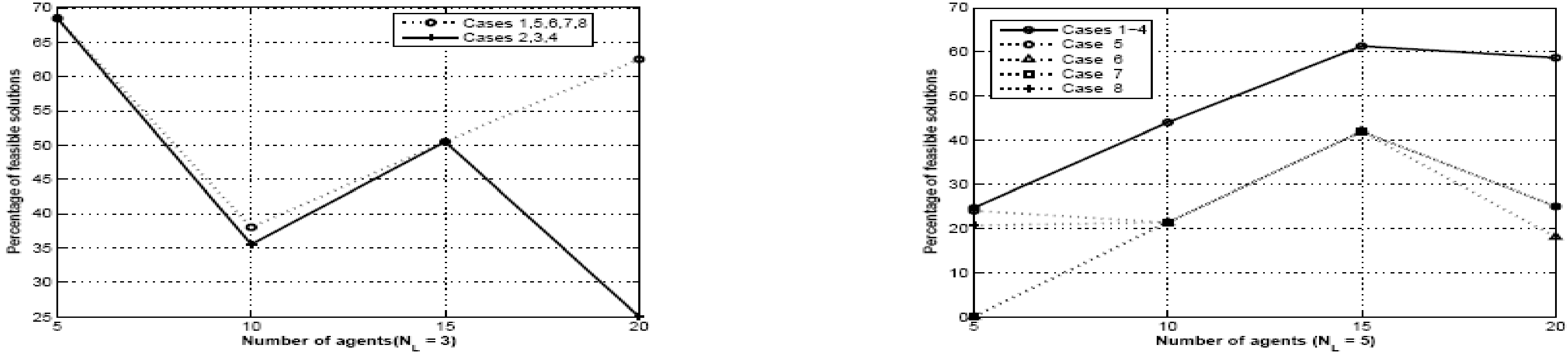
Learning

- For each decision variable x_{ij} two neurons are used: one to represent the "1" and the other the "0".
- We experiment with two different RNNSI architectures:
  1. Architecture 1 (cases 1-4): Neurons are fully interconnected in terms of W^+ and W^-
  2. Architecture 2 (cases 5-8): Neurons excite other neurons of the same polarity and inhibit neurons of the opposite (see the Figure to the right)
- Different cases corresponding to different approaches to the A weights
- We train the network with 200 randomly generated optimisation problems with varying I_j, the other parameters being constant. The optimal solution is computed in each case and used to train the RNNSI

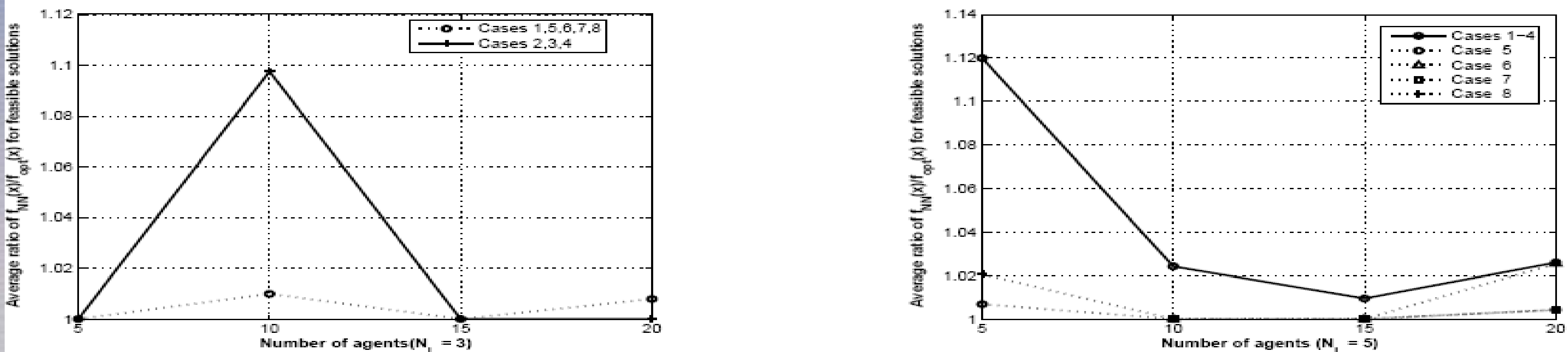


Imperial College London

200 examples were generated at random to test the RNNSI:



Percentage of feasible solutions for $N_L = 3$ and $N_L = 5$



Average ratio of $f_{NN}(\underline{x})/f_{opt}(\underline{x})$ for the feasible solutions for $N_L = 3$ and $N_L = 5$

Summary of the Approach

- Off-line learning is used to “train” a neural network so as to serve as an “Oracle” providing fast, distributed, consistent and accurate response to an optimisation problem that would normally be NP-hard and therefore not solvable in real-time
- The trained neural network is given to all agents (or emergency units).
- When the emergency happens, identical information about the emergency (whereabouts of the incidents, estimate of the number of victims) is broadcast to all agents
- Each individual agent uses its Oracle to obtain fast, distributed, and consistent decisions
- Since all agents have the same “Oracle”, if they have the same information there will be no conflicts in their decisions
- The approach should be robust to small uncertainties in parameters: i.e. small errors in the data that is broadcast, or small differences between the data received by different agents

Further Work

- Investigate the robustness of the approach to errors in the data, and to differences between the data received by different agents – both theoretically and numerically
- Study different cost functions that may better reflect the needs of the application
- Integrate the approach into the RT4 simulators and vignettes – ROBOCUP and BES
- Research on decisions that are composed of multiple stages – e.g. not just the allocation of the ambulance but also the route it must take
- Consider more realistic “error functions” for the RNNSI .. Not just quadratic
- Research on coupled or synchronised decisions
- Study other methods for distributed decision making such as auctions, coalitions .. and compare the results with this work

Further Work

Especially for the Multistage Problem

- Examine the relationship of the two formulations to understand when one formulation is better than the other.
- Develop real-time heuristic algorithms (e.g. maximum execution time 1s)
- Device partitioning methods so that the problem can be divided into smaller easier to solve problems
- Develop distributed algorithms and compare the performance (speed, efficiency) with the optimal
- Test the performance of the RNNSI learning algorithm on the particular problem

Imperial College London

G-Networks and Gene Regulatory Networks

PHYSICAL REVIEW E **76**, 1 (2007)

Steady-state solution of probabilistic gene regulatory networks

Erol Gelenbe*
Department of Electrical and Electronic Engineering, Imperial College, London SW7 2BT, United Kingdom
(Received 27 February 2007; revised manuscript received 10 July 2007)

We introduce a probability model for gene regulatory networks, based on a system of Chapman-Kolmogorov equations that represent the dynamics of the concentration levels of each agent in the network. This unifying approach includes the representation of excitatory and inhibitory interactions between agents, second-order interactions which allow any two agents to jointly act on other agents, and Boolean dependencies between agents. The probability model represents the concentration or quantity of each agent, and we obtain the equilibrium solution for the joint probability distribution of each of the concentrations. The result is an exact solution in "product form," where the joint equilibrium probability distribution of the concentration for each gene is the product of the marginal distribution for each of the concentrations. The analysis we present yields the probability distribution of the concentration or quantity of all of the agents in a network that includes both logical dependencies and excitatory-inhibitory relationships between agents.

DOI: XXXX PACS number(s): 87.16.Ac, 87.16.Yc, 87.17.−d

Imperial College
London

Outline

Rene' Thomas' web page:

``Most biological regulatory systems involve complex networks of interactions. Theoretical modelling, together with simulations and computational approaches, provides a useful framework for integrating data and gaining insights into the dynamical and functional properties of such networks.

In this perspective, a major aim of the research is to contribute to the understanding of how regulatory mechanisms at various scales (e.g. molecular, cellular and intercellular) act synergistically or competitively to achieve degrees of regulation not attainable by one mechanism alone.

Key issues are the variety of attractors possible for a network, the nature of transition states and transition dynamics, and the role of the network in emergent behaviour.

These issues are examined in terms of systems of differential equations, automata networks and probabilistic models."

Imperial College London

A Regulatory Network

- **A set of nodes representing "genes"**
- **Associated with each node, a non-negative real or integer number representing the "level of concentration" of the gene**
- **Directed arcs between nodes representing the interactions between genes**
- **Arcs are labeled with {+,-}/{excitation,inhibition}**
- **Arcs are also labeled with thresholds, i.e.**

$$[u](x) \xrightarrow{+,\ z} [v](y)$$

[u] activates [v] if $x \geq z$

A Regulatory Network

- Arcs are also labeled with thresholds, i.e.

[u](x) → [v](y)

+, z

If [u] is active, it [u] activates [v] if $x \geq z$

Dynamics:

A node is “active” if all of its predecessors with excitation conditions are also active

It is inactive if at least one of its predecessors with excitation condition is inactive, or when one of its predecessors with inhibition condition is active

Synchronous or asynchronous time may determine and change the state of each node sequentially

Concentration levels:

We may also label nodes with concentration levels which themselves vary with the dynamics
In this case we are not just interested in the “activation” but also in the “level of activation”

Probability Model

- Notation: Agents or Genes {1, …, n}, t is time

$K(t) = (K_1(t), \ldots , K_n(t))$ are the concentration levels of the "genes"

$x_1, \ldots , x_n$ are the activation thresholds of the "genes"

Gene j is active if $K_j(t) > x_j(t)$

If a gene or node is "active" it may contribute to activate or disactivate any one of its successors

More generally we would like to represent all Boolean dependencies between agents

As we consider continuous time, it is reasonable to assume dynamics where "one thing happens at a time" in some very small interval of time $[t, t+\Delta t]$

We develop a formalism that allows us to write equations for the probabilities

$P[K,t]$ = Probability[$K(t) = K$ | Initial conditions at $t=0$]

and hence for $P[A(t) = A$ | Initial conditions at $t=0]$

Imperial College London

The G-Network

- The $K_i(t) \geq 0$. These are integer valued random variables which represent the concentration or quantity of the agents i at time $t \geq 0$.

- $\Lambda_i \geq 0$ is a real number representing the rate at which agent i is being replenished from some external source; similarly $\lambda_i \geq 0$ is the rate at which agent i is being depleted, provided that agent i is present in some concentration.

- The $r_i \geq 0$ are real numbers representing the activity rates of each agent i, provided again that the agent is present in some non-zero amount. In precise terms, λ_i, λ_i, r_i are the parameters of exponential distributions, and Λ_i, λ_i are the arrival rates of independent Poisson processes of signals which, respectively, increase or decrease the level of the variable $K_i(t)$. Similarly r_i is the average time between successive interactions of agent i with other agents.

$$
\begin{aligned}
w^+(i,j) &= r_i P^+(i,j) \\
w^-(i,j) &= r_i P^-(i,j) \\
w(i,j,l) &= r_i Q(i,j,l).
\end{aligned}
$$

- The parameter $K_i(t)$ represents the *activation level* of the agent i. If $K_i(t) > 0$ the agent is activated, and $K_i(t)$ can also be used to represent the amount or concentration of agent i that is present at time t. The equations (6) represent the case where the agent i is only activated if $K_i(t) \geq x_i \geq 1$.

- Through the parameters Λ_i, the natural replenishment of agent i, for instance via some biochemical reaction, or via infiltration from an external medium, is being represented.

- The parameters λ_i in turn represent a deletion of agent i. Both Λ_i and λ_i are specific to a single agent and do not represent inter-agent interactions.

- The parameters r_i represent the depletion of agents i as a result of the agents' interaction with other agents, or via removal of the agents from the medium being considered through the rates $r_i d_i$. Note that $r_i = \sum_{j=1}^{n}[w^+(i,j) + w^-(i,j) + \sum_{l=1}^{n} w(i,j,l)]$.

- The parameters $w^+(i,j)$ and $w^-(i,j)$ represent the replenishment or depletion of agent j, or the excitation or inhibition effect, as a result of agent i.

- Finally, the parameters $w(i,j,l)$ represent the excitation/activation of agent l through the effect of i and j, or the rate of increase of the amount of l through the effect of i, j.

Thus the G-network will represent both the inter-agent relations with respect to activations, and/or the amounts or concentrations of the agents and the manner in which this affects their interactions and activation.

G-Network Dynamics and Stationary Solution

$$
\begin{aligned}
\frac{dP(k,t)}{dt} &= \sum_{i=1}^{n} [\, P(k+e_i,t)(\lambda_i + r_i d_i) + \Lambda_i P(k-e_i,t)1[k_i>0] - P(k,t)[\Lambda_i + 1[k_i>0](\lambda_i + r_i)] \\
&+ \sum_{j=1}^{n} [\, 1[k_i+1 \geq x_i](P(k+e_i-e_j,t)1[k_j>0]w^{+}(i,j) + P(k+e_i+e_j,t)w^{-}(i,j) \qquad (6) \\
&+ \sum_{l=1}^{n} P(k+e_i+e_j-e_l,t)1[k_i+1 \geq x_i]1[k_j+e_j \geq x_j](w(i,j,l)+w(j,i,l)) \,]\,]
\end{aligned}
$$

Let $P(k) = \lim_{t\to\infty} P(k,t)$ and consider the equations (6) in which we have set $x_i = 1,\ i = 1, \ldots, n$. In other words, as long as there is at least one agent of type i, agent i is activated. Let us introduce the term:

$$q_i = min[1, \frac{\Lambda_i + \sum_{j=1}^{n} q_j w^{+}(j,i) + \sum_{j,l=1}^{n} q_j q_l w(j,l,i)}{r_i + \lambda_i + \sum_{j=1}^{n} q_j w^{-}(j,i) + \sum_{j,l=1}^{n} q_l w(l,i,j)]}],\ i = 1, \ldots, n \qquad (7)$$

which represents the probability that agent i is activated.

Theorem 1 Consider the case where $x_i = 1,\ i = 1, \ldots, n$. For any subset $I \subset \{1, \ldots, n\}$ such that $q_m < 1$ for each $m \in I$, and $I = \{m_1, \ldots m_{|I|}\}$:

$$
\begin{aligned}
P(K_m = k_m) &= q_m^{k_m}(1-q_m),\ and \qquad (8) \\
P(K_{m_1}, \ldots, K_{m_{|I|}} = k_{m_1}, \ldots, k_{m_{|I|}}) &= \Pi_{i=1}^{|I|} q_{m_i}^{k_{m_i}}(1-q_{m_i}) \qquad (9)
\end{aligned}
$$

Theorem 2 If all $x_i = 1$, the solution of (6) with $x = (1, \ldots, 1)$ as provided by (7), (9), (9) exists and is unique.

Logical Interactions of Agents

- Introduce a set of "dummy agents" $A_1, A_2, \ldots, A_\alpha$ that act as intermediaries between the set of agents a.
- a_1 acts upon A_1, (a_2, A_1) act upon A_2 and so on. Finally $(a_\alpha, A_{\alpha-1})$ act upon A_α, and A_α acts upon agent l in an excitatory manner with $w^+(A_\alpha, l) = 1$.
- Furthermore we set $\Lambda_{A_s} = \lambda_{A_s} = 0$, $r_{A_s} = 1$ for $1 \le s \le \alpha$, $w(a_1, a_2, A_2) = 1$ and $w(a_s, A_{s-1}, A_s) = 1$ for $s = 3, \ldots, \alpha$.
- We also introduce dummy agents $B_1, \ldots, B_\beta$ so that (b_1) acts upon B_1 in an excitatory manner with $w^+(b_1, B_1) = 1$, (b_2) acts upon B_2 similarly, and so on, and b_β acts upon B_β in an excitatory manner with B_β with $w^+(b_\beta, B_\beta) = 1$.
- Then each B_s acts upon agent l in an *inhibitory* manner with $w^-(B_s, l) = \gamma$, $1 \le s \le \beta$.
- We set $\Lambda_{B_s} = \lambda_{B_s} = 0$, $r_{B_s} = 1$ for $1 \le s \le \beta$.

$$Agent_l = [\wedge_{s=1}^{\alpha}(Agent_{a_s})] \bigwedge [\wedge_{s=1}^{\beta}(\neg Agent_{b_s})],$$

$$q_l = min[1, \frac{\prod_{s=1}^{\alpha} q_{a_s}}{\sum_{s=1}^{\beta} \gamma q_{b_s}}]$$

Imperial College London

Obtaining the Complement – Exact Approach
$A_l = U\ \{\Pi A_i\ \Pi[\text{not } A_j]\}$

Turning to expression (7), we see that for any q_i the term $\rho_i = [1 - q_i]$ is:

$$\rho_i = \frac{r_i + \lambda_i - \Lambda_i + \sum_{j=1}^{n} q_j[w^-(j,i) - w^+(j,i)] + \sum_{j,l=1,\ l\neq j}^{n} q_l[w(l,i,j) - q_j w(l,j,i)]}{r_i + \lambda_i + \sum_{j=1}^{n} q_j w^-(j,i) + \sum_{j,l=1,\ l\neq j}^{n} q_l w(l,i,j)]}, \quad (33)$$

Note that we would like to have an agent, say $Agent_{c_i}$, whose state is the complement of agent $Agent_i$ so that ρ_i is the stationary distribution that $Agent_{c_i}$ is activated. Thus we require that the parameters in the expression (33) have the following properties:

If $Agent_i$ has, in the same network, a complementary agent A_{c_i}, then:

$$\rho_i = \frac{L_i + \sum_{j=1}^{n} q_j \Omega^+(j,i) + \sum_{j,l=1,\ l\neq j}^{n} q_j q_l \Omega(l,j,i)}{R_i + l_i + \sum_{j=1}^{n} q_j \Omega^-(j,i) + \sum_{j,l=1,\ l\neq j}^{n} q_l \Omega(l,i,j)}, \quad (34)$$

with

$$\begin{array}{rl}
(I) & L_i = r_i + \lambda_i \geq \Lambda_i \\
(II) & \Omega^+(j,i) = w^-(j,i) - w^+(j,i)\ \geq 0\ \ for\ \ any\ j \neq i \\
(III) & \Omega^-(j,i) = w^-(j,i)\ \ for\ \ any\ j \neq i \\
(IV) & w(l,i,j) > 0 \Rightarrow w(l,j,i) = 0\ for\ any\ l,j \neq i \\
(V) & w(l,i,j) = 0 \Rightarrow w(l,j,i) = 0\ for\ any\ l,j \neq i \\
(VI) & r_i + \lambda_i - \Lambda_i \geq 0
\end{array} \quad (35)$$

$$\begin{array}{rl}
(II) & \Rightarrow \Omega^+(j,i) = w^-(j,i) - w^+(j,i)\ for\ all\ j \neq i \\
(IV)\ and\ (V) & \Rightarrow \Omega(l,i,j) = \Omega(l,j,i) = 0\ for\ all\ l,j \neq i
\end{array}$$

Imperial College London

The G-Network model provides the structure to model

Boolean dependencies between agents in

Conjunctive Normal Form

$$A_F = U \{\Pi A_i \, \Pi[\text{not } A_j]\}$$
$$q_F = \Sigma \{ \Pi q_i \, \Pi \rho_j\}$$

Toy Example of four agents $\{A_0, A_1, A_2, A_3\}$
A_i inhibits $[A_{(i+1)mod4}$ and $A_{(i+2)mod4}]$

Interpretation 1: $q = 1/(1+2q) = 0.5$
Interpretation 2: $q = (1-q)(1-q) = 0.382$

Thus the "semantics" we associate with a regulatory network model has to be precisely indicated so as to derive the appropriate probabilistic

Imperial College London

Computing the Logical Dependencies in Gene Regulatory Networks

Consider the Boolean function $F:[0,1]^n \rightarrow [0,1]$ in CNF, written as the conjunction of disjoint terms

$$F = \bigvee_{u=1}^{m} \tau_u, \qquad (32)$$

where we have the following:

(i) A term is written as $\tau_u = X_{u1} \wedge \cdots \wedge X_{u\alpha}$ with X_{us} being either B_{a_s} or $B_{a_{cs}} = \neg B_{a_s}$,

$$q_F = \sum_{u=1}^{m} \prod_{s \in \Upsilon_u}^{\alpha} q_{a_s} \prod_{s \in \Phi_u} [1 - q_{a_s}],$$

Theorem 5. For any expression in CNF (32), there exists a G network with a set of agents A, which contains the agent F, the agents $\{a_1, \ldots, a_\alpha\}$ and their complements $\{a_{c1}, \ldots, a_{c\alpha}\}$, as well as dummy agents $\{A_{us}: 1 \leqslant u \leqslant 2^\alpha, 1 \leqslant s \leqslant \alpha\}$, such that for $q_F = \lim_{t\to\infty} P[B_F(t)=1]$ is given by (33).

Before we detail the proof let us indicate that this result states that, given a specified Boolean dependency between agents of a regulatory network, one can use the G-network model to represent these Boolean dependencies. Since the regulatory network itself is *probabilistic*, these Boolean dependencies will be reflected in equations about the *probabilities of the state of the agents*, i.e., these probabilities will be are consistent with the Boolean dependencies that have been given.

Proof of Theorem 5. From Sec. IV A we know that

$$q_{a_s} = \frac{\Lambda_{a_s} + \sum_{jA} w^+(j,a_s)}{r_{a_s} + \lambda_{a_s} + w^-(j,a_s)},$$

$$q_{a_{cs}} = \frac{r_{a_s} + \lambda_{a_s} - \Lambda_{a_s} + \sum_{jA} q_j[w^-(j,a_s) - w^+(j,a_s)]}{r_{a_s} + \lambda_{a_s} + \sum_{jA} w^-(j,a_s)}.$$

Imperial College London

Electronic Network <-> Random Neural Network
Future Work: Back to our Origins

- Very Lower Power Ultrafast “Pseudo-Digital” Electronics
- Network of interconnected probabilistic circuits
- Only pulsed signals with negative or positive polarity
- Integrate and fire circuit = Neuron [RC circuit at input, followed by transistor, followed by monostable]
- When RC circuit’s output voltage exceeds a threshold, the “Neuron’s” output pulse train is a sequence of pulses at the characteristic spiking rate (μ) of the neuron
- Frequency dividers (eg flip flops) create appropriate pulse trains that emulate the appropriate neural network weights
- Threshold circuits (eg biased diodes and inverters) create appropriate positive or negative pulse trains for different connections

Imperial College London

Micro-Economics

Auctions in Networks

Imperial College London

Vision: The World's Economy will be governed by Electronic Economic Transactions in Cognitive Networks

Auction: Formal Mechanism that Governs Decisions for Economic Transactions or Resource Allocation and Exchange

Cognitive Network: A Computer-Communication Network where Resource Allocation including Routing is Achieved by Adaptive Procedures that Optimise QoS, Profit or Other Criteria

Imperial College London

Vision

The World's Economy is a "Chain Reaction" of Electronic Economic Transactions

Users and Services, Buyers and Sellers, are Agents with Interchangeable Roles

Computer Networks are the Infrastructure of the World Economy

Networks are Becoming Autonomic and Cognitive

Auctions – Economic mechanisms which have been studied both in Economic Theory and in Computer Science

- Guo, X. 2002. An optimal strategy for sellers in an online auction. ACM Trans. Internet Tech. 2 (1): 1–13.

- Hajiaghayi, M. T., Kleinberg, R., and Parkes, D. C. 2004. Adaptive limited-supply online auctions. Proc. 5th ACM Conference on Electronic Commerce, May 17-20, 71–90.

- McAfee, R. P. and McMillan, J. 1987. Auctions and bidding. J. Economic Literature 25: 699–738.

- Milgrom, P. R. and Weber, R. 1982. A theory of auctions and competitive bidding. Econometrica 50: 1089–1122.

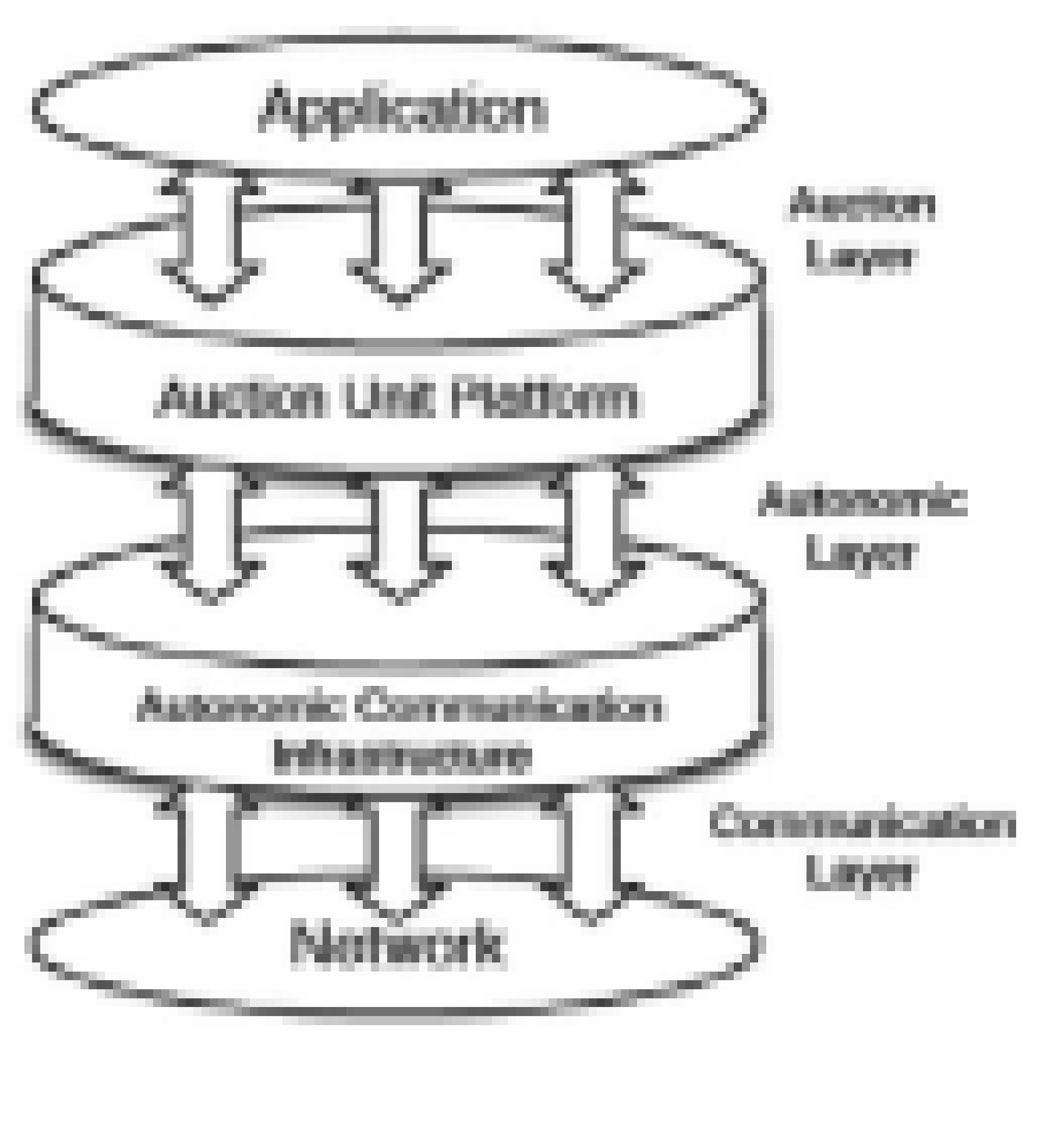
- Shehory, O. 2002. Optimal bidding in multiple concurrent auctions. International Journal of Cooperative Information Systems 11 (3-4): 315–327.

- Gelenbe, E. 2008. Networked Auctions. In press ACM Trans. Internet Tech.

Imperial College London

Auctions

Cognitive Networks



Imperial College London

Auctions have Many Buyers (Bidders) and Sellers – Leading to Interesting Research Questions

- Mechanisms that create incentives: encouraging certain behaviours
- Outcome of collective behaviours
- Coordinated buying and selling (cartels, trusts)
- Rational Bidders
- Adversarial Behaviours (competitors)
- Learning from collective behaviour: observing the buyers and sellers

- Modelling collective behaviour
- Effect of Network QoS on Economic Considerations
- Malicious Behaviours & Protecting the Information Infrastructure

Imperial College London

Modeling Auctions

- Decision Framework: An auction System
- The value of the good for the bidders is a r.v. V, whose prob. distribution p(v)=P[V=v], which may be unknown to the seller
- Buyers will not bid above the value they associate with the good, but V=infinity is possible (.. I am willing to buy it at any price ..)

- The seller observes the bids, and after each bid waits for some time before accepting the bid; a new bid may arrive in the meanwhile, and the process repeats itself
- How and when should the seller accept the bid?
- What is the probabilistic outcome of such a system, in terms of the expected price that the good brings in, or the time it takes to sell the good, or the income generated per unit time?
- How can the seller learn the value of the good and act accordingly?
- How can bidders also adapt their behaviour to get the best price?
- How can buyers take advantage of multiple Networked Auctions?

Imperial College London

The Secretary or Sultan's Dowry Problem .. Related but Different
Martin Gardner, Scientific American, 1960:

- A sequence of candidates show up, each of value or quality C_1, C_2, .. , C_n, .. which are r.v.'s
- The buyer's purpose is to select one of these whose quality is close to the maximum quality
- The buyer observes the sequence for a finite time, hoping to wait long enough to select the best .. and selects the k-th, after which the decision is irrevocable
- What is the probability that the one selected is the optimum?
- The outcome will select the best with probability 1/e
- Y.S. Chow et al., Israel J. Math. 2, 81-90, 1964
- S.R. Finch "Optimal stopping constants", in Mathematical Constants, Cambridge Univ. Press, 361-363, 2003

Imperial College London

An Auction

For a fixed value v of the "good" we have a state space $\{0,1, \ldots ,v,A_1, \ldots , A_v\}$ where i represents the value that is attained after the i-th bid, while A_i is the state entered after the i-th bid is accepted – the 0-state is a "rest state" after a particular auction is complete

The random process representing the current state of, or value attained by, the auction is Y_t ; alternately X_t is the index of the bid and $f(Y_t)$ may be the monetary value

$0<t_1< \ldots < t_n < \ldots$, are instants at which the auction starts, with auction end times at $t_n+E_n< t_{n+1}$ when the buyer accepts the offer, and rest times $R_n=t_{n+1}- [t_n+E_n]$

Imperial College London

Increments offered by successive bidders within any one auction are random variables $X_1, \ldots, X_i, \ldots$ that may depend on the auction number n .. X_{ni} etc.

The time between the arrival of successive bids are r.v. $\{T_{ni}\}$

After bid "ni" is received, the seller will wait some "think" or decision time D_{ni}, after which it will accept the bid if $D_{ni} \leq T_{n,i+1}$, or consider the next bid if $D_{ni} > T_{n,i+1}$ unless a new bid arrives

If there is reneging or balking, the most recent bid may be revoked after some time B_{ni} if $B_{ni} < D_{ni}$. If the bid is revoked by the highest bidder, then the next highest un-revoked bid becomes the valid bid, and may also be reneged, etc.

Imperial College London

The price attained during the n-th auction will then be the r.v.

$$Q_n = \Sigma^{N(n)}_{i=1} X_{ni}$$

Furthermore bids may be a function of the value attained by the good during the preceding bid, e.g.

$$X_{n, K+1} = g_{n, K+1}(\Sigma^{k}_{i=1} X_{n\ i})$$

or a function of the value V_n of the good as well, e.g.

$X_{n, K+1} = g_{n, K+1}(U_n , \Sigma^{k}_{i=1} X_{n\ i})$, or more specifically

$$X_{n, K+1} = g_{n, K+1}(U_n - \Sigma^{k}_{i=1} X_{n\ i})$$

Analytical Results

Bids arrive to an auction according to a Poisson process; $1/\lambda$ is the average time between successive bids

- $1/\delta$ is the average time that the seller waits before accepting a bid (possible decision variable) – the corresponding time is an exponentially distributed r.v.

- 1/r is the average rest period after the end of an auction and before the next auction restarts. Without loss of generality r=1; this time can have a general distribution

- Assume there is no balking or reneging

- The value of the good is fixed to a given r.v. V with arbitrary distribution function, identical at each successive auction

- Then after analysis

$$E[\text{ Sale price } | \ V=v\ ] = [1-\rho^v]/[1-\rho] \leq v, \quad \rho = \lambda/(\lambda+\delta)$$

$$E[\text{ Income per unit time }] = (1- E[\rho^V]\ )\ \lambda r(\lambda+\delta)/(\lambda r+\lambda\delta+r\delta)$$

Imperial College London

The results generalize to iid bid sizes, and to other models in which a Markov renewal structure can be exploited

$$E[\text{Sale price} \mid V=v] = E[X]\,[1-\rho^{v}]/[1-\rho] \leq vE[X],$$

$$\rho = \lambda/(\lambda+\delta)$$

$$E[\text{Sale price} \mid V=v] = [1-\rho^{v}]/[1-\rho]\,\Sigma^{V}_{I=1} X_I$$

$$E[\text{Income per unit time}] = E[X](1-E[\rho^{V}])\lambda r(\lambda+\delta)/(\lambda r+\lambda\delta+r\delta)$$

For the Vickrey auction where the good is sold to the highest bidder at the second highest price:

$$E[\text{Sale price} \mid V=v] = E[X]\,\rho\{[1-\rho^{v-1}]/[1-\rho]+\delta/\lambda\}$$

Imperial College London

Auctions with a minimum sale price s:

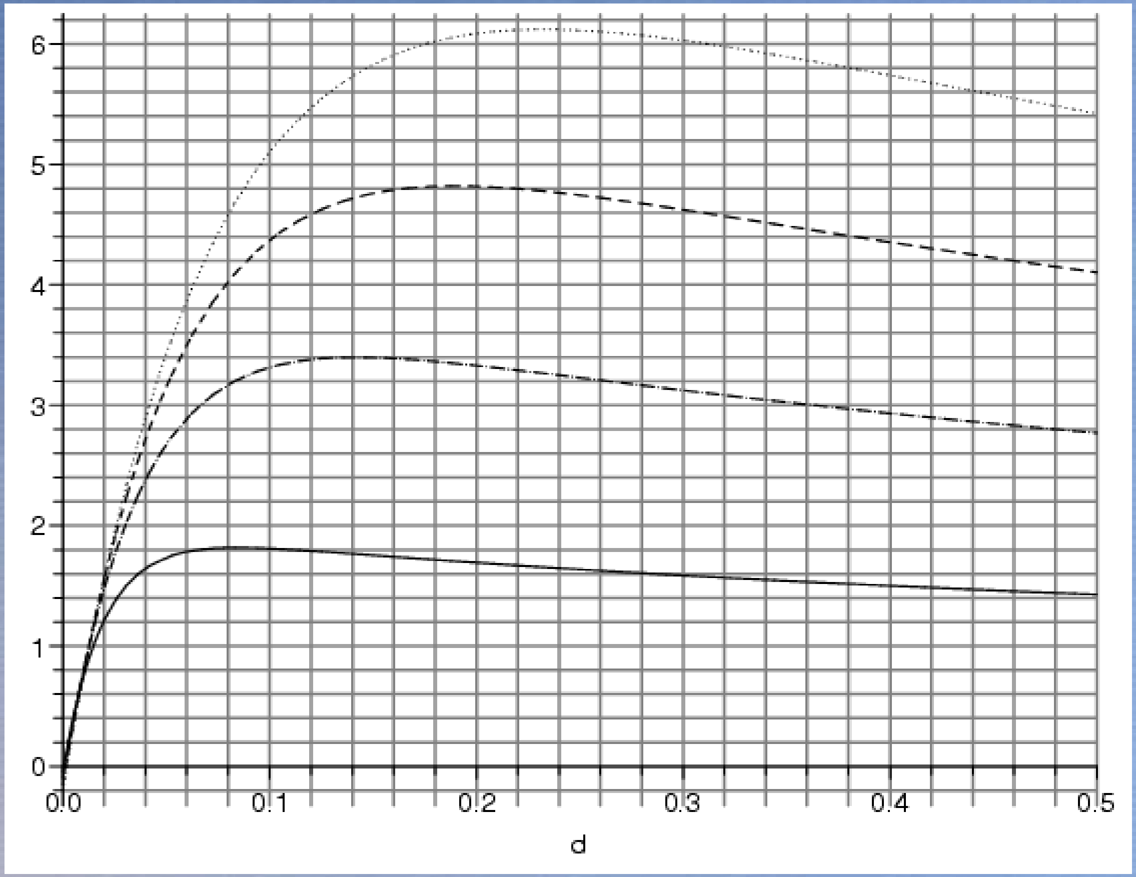
$$E[\text{Sale price} \mid V=v] = E[X]\,\rho\,\{\lambda_s\,[1-\rho^{v-s}]/\delta + s\}$$

where $\rho = \lambda_s / (\lambda_s + \delta)$

When the un-successful bidder re-bids with probability p and new bidders arive at rate γ:

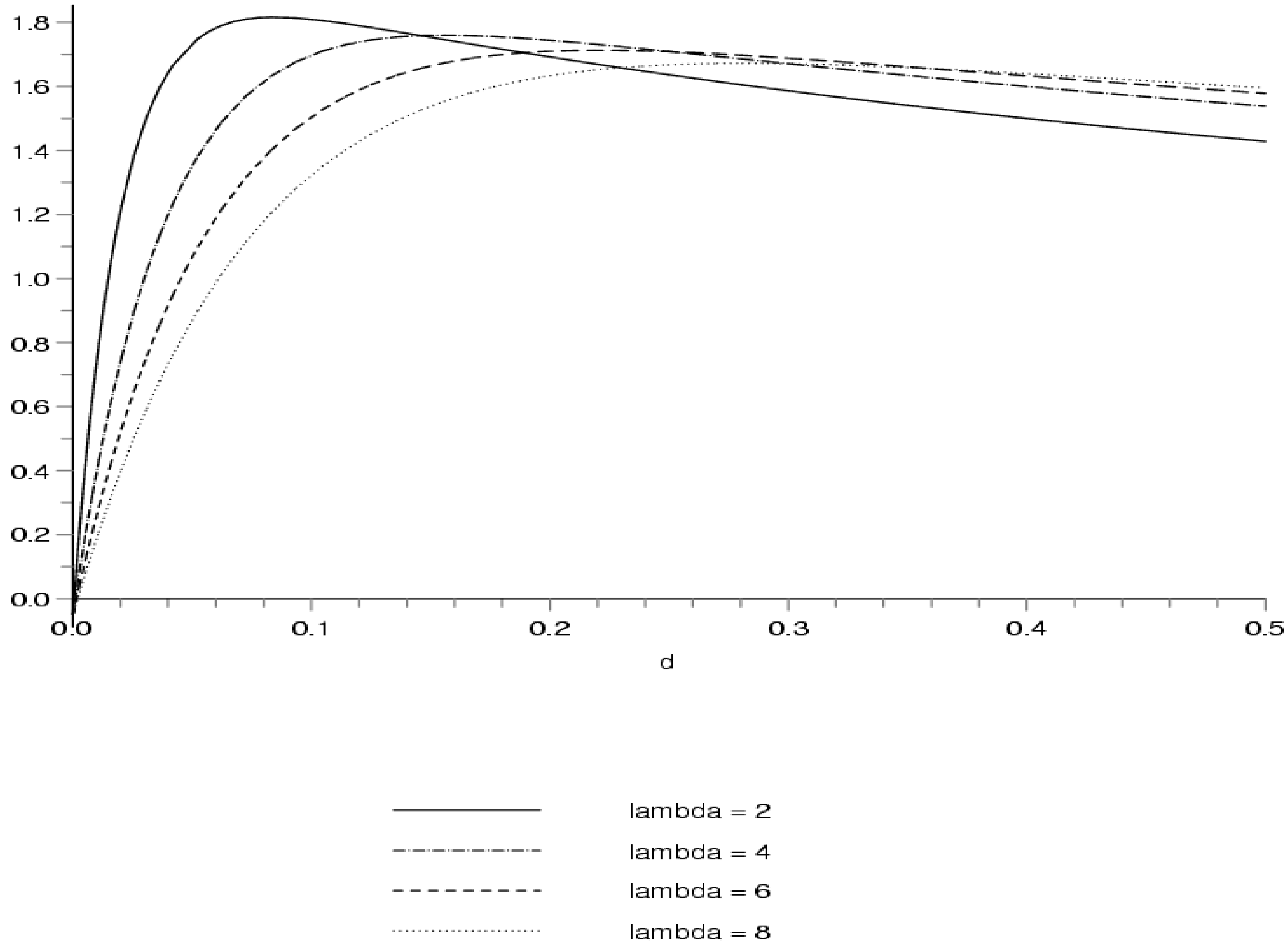
$$\lambda = \gamma + \lambda\, p\, [\Phi - 1]/\,\Phi = \gamma\,[1-p+p/\,\Phi]$$

Imperial College London



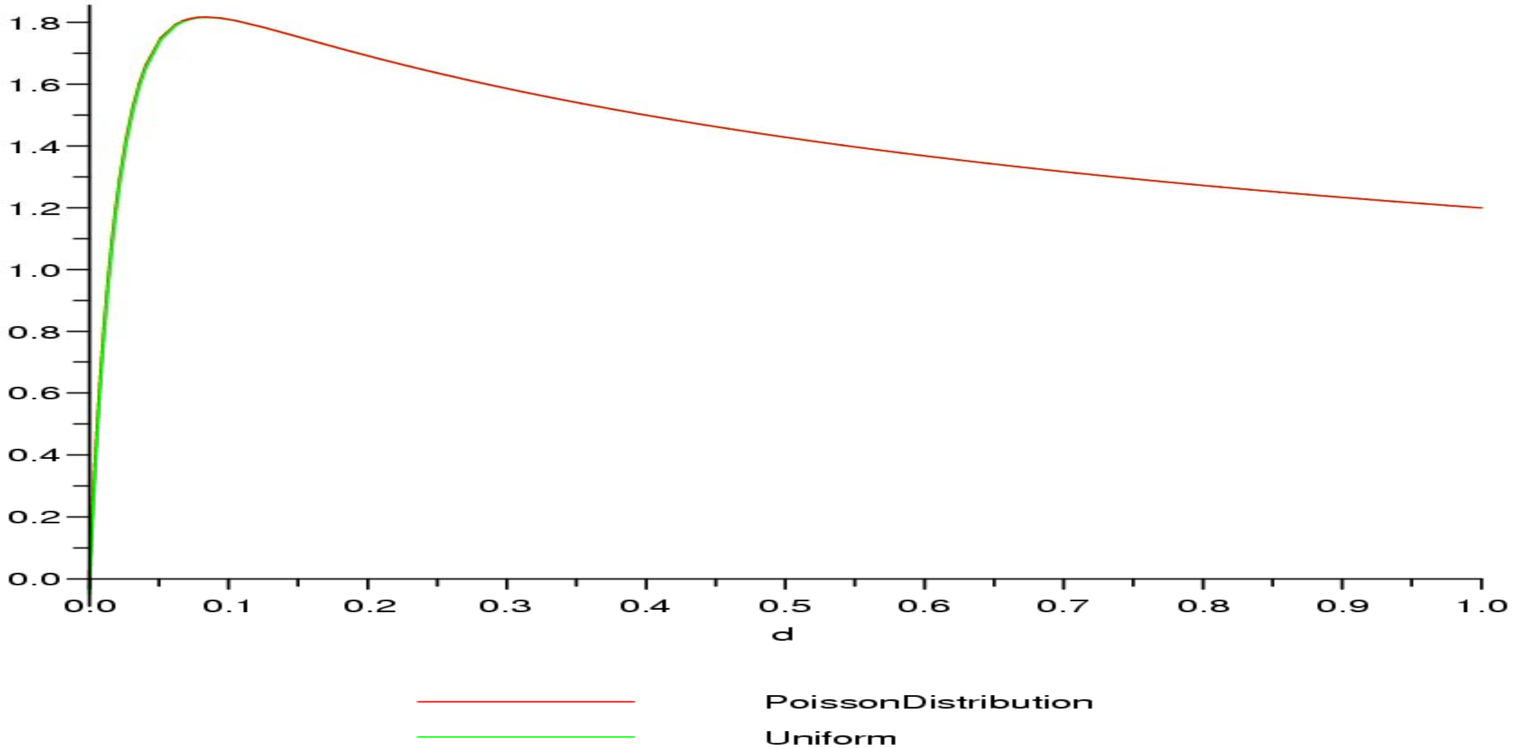
Income per unit time vs rate δ at which decisions are made for a high 8 down to low 2 (bottom) rate at which bids arrive. The value of the good Is uniformly distributed between 80 and 100 units

Imperial College London



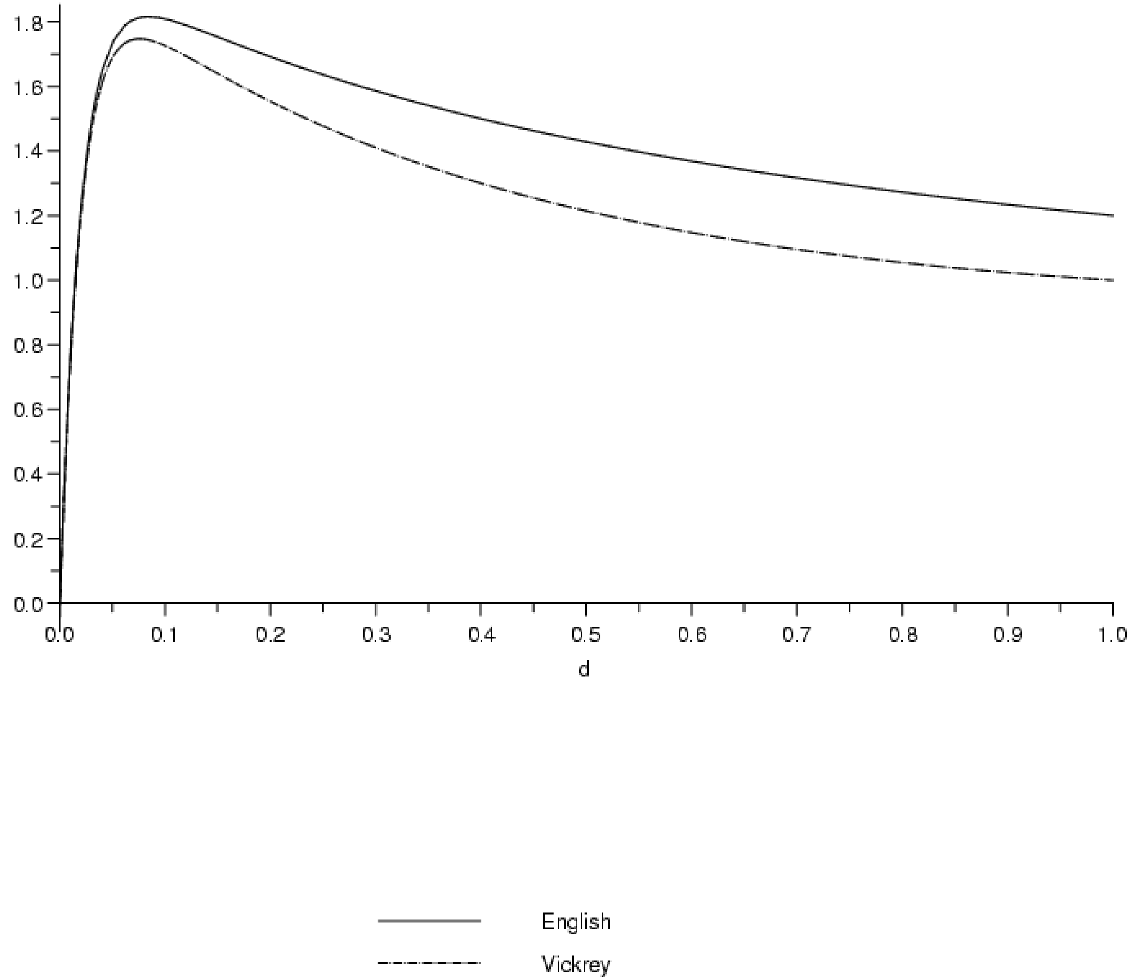
Income per unit time vs rate δ at which decisions are made for an English auction with unit increments: comparison of the effect of the arrival rate of bids. The value of the good is uniformly distributed between 80 and 100 units

Imperial College London

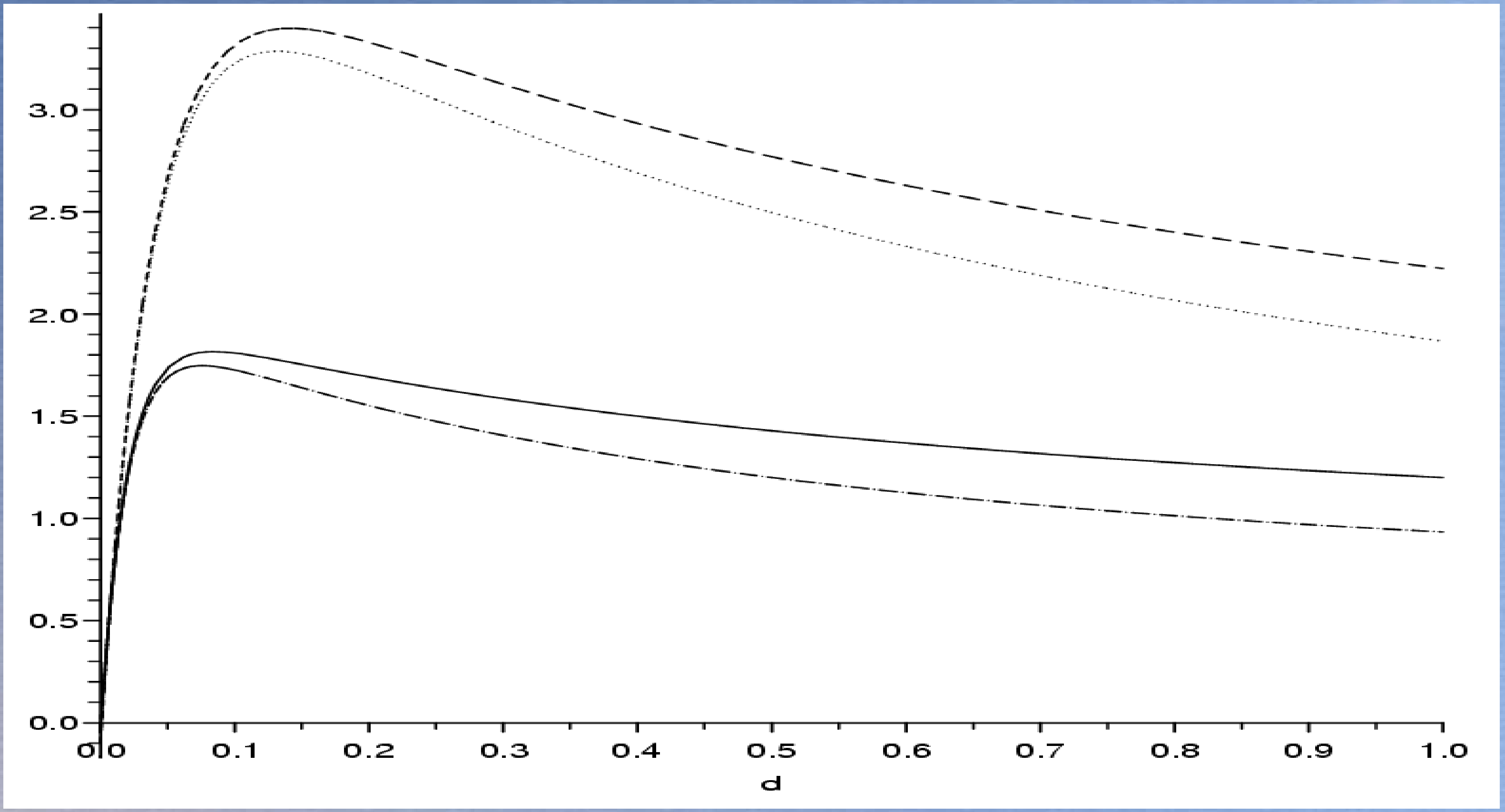


Comparison of the effect of a uniformly distributed and Poisson distributed value of the good on the Income per Unit Time

Imperial College London

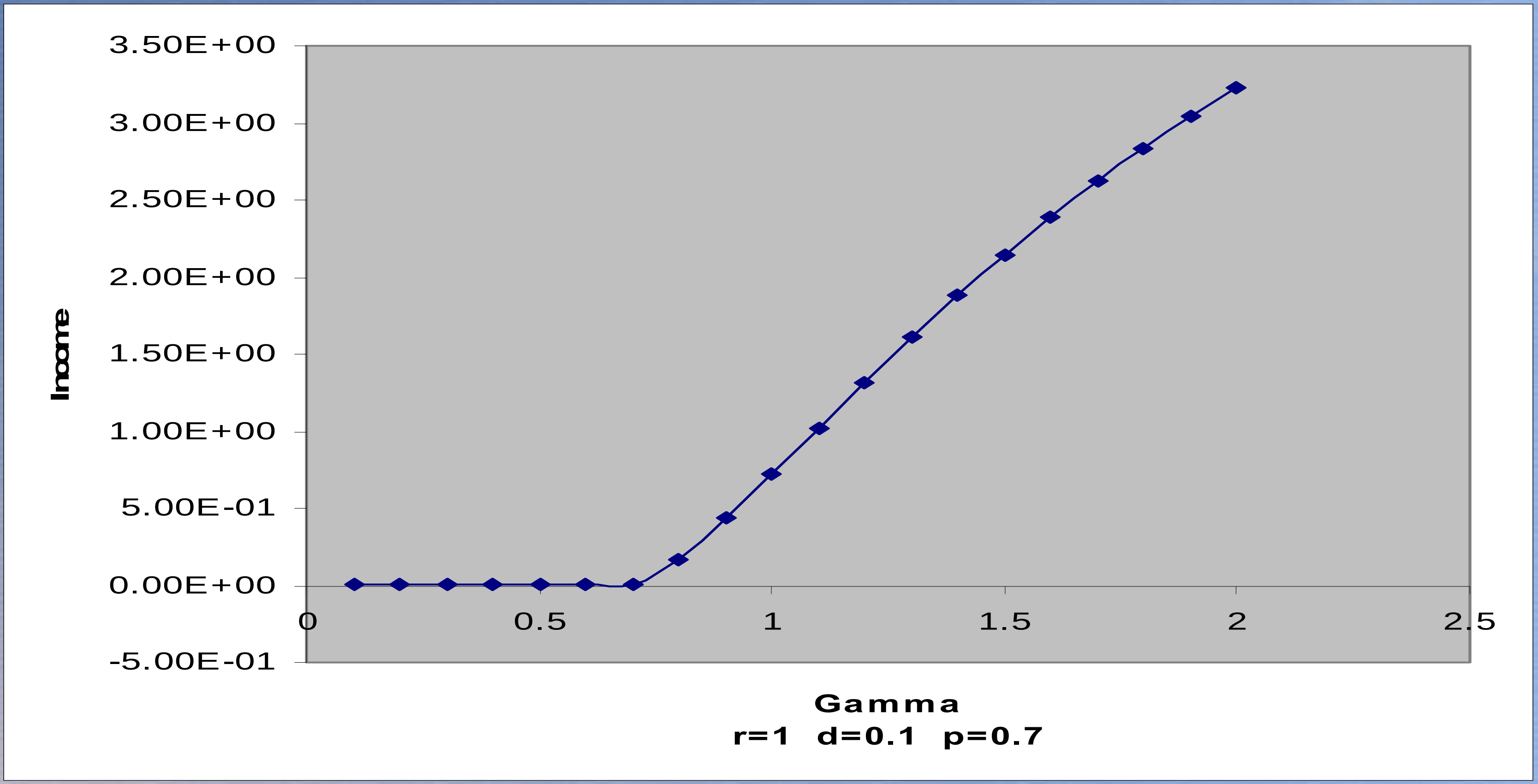


Imperial College London



Income per unit time vs rate δ at which decisions are made for the English and Vickrey auctions with unit increments. The arrival rate of bids is 4 (above) and 2 (bottom). The value of the good Is uniformly distributed between 80 and 100 units

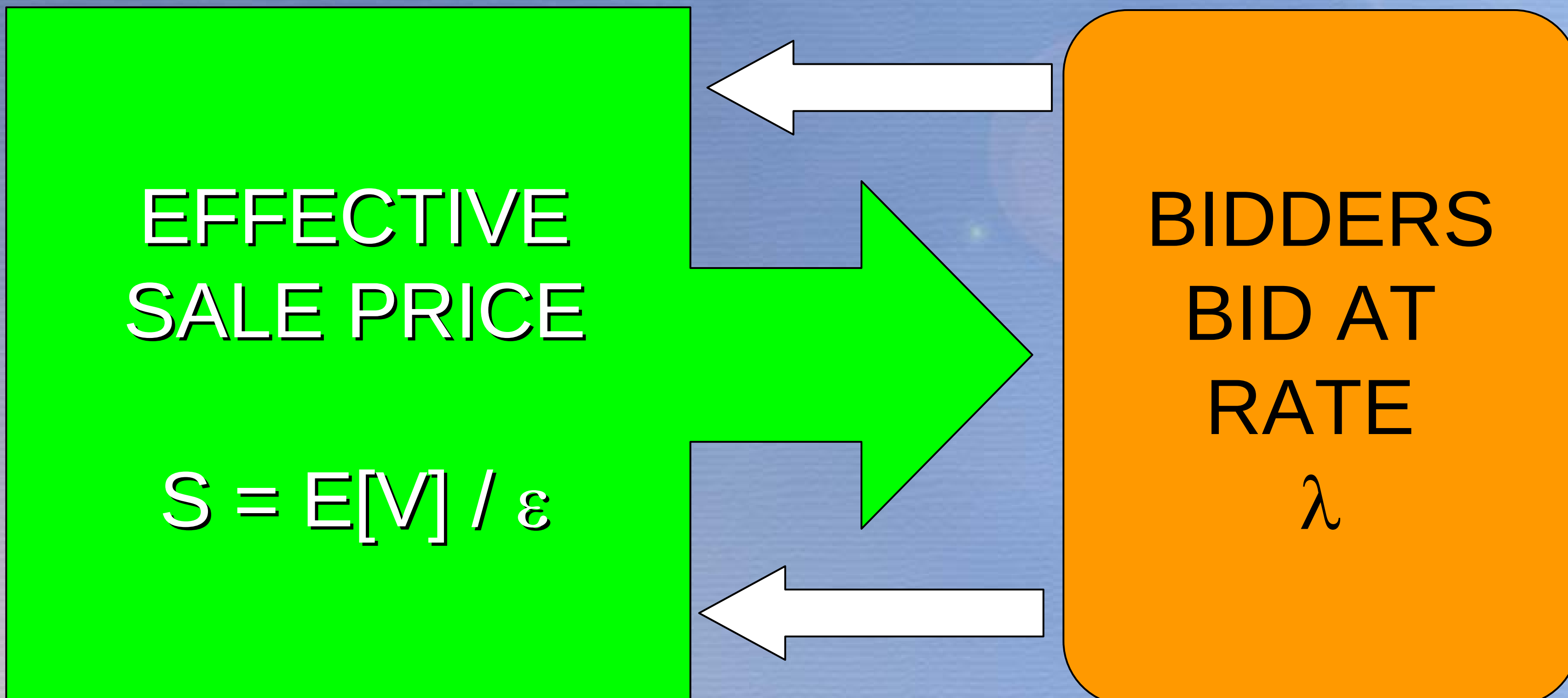
Imperial College London



Income per unit time vs the rate at which bids arrive for δ=0.1, and r=1 with the probability that an unsuccessful bidder will try again p=0.7

Imperial College London

Smart Price Formation



Imperial College London

Smart Price Formation by Watching the Market

S_v = E[Sale price | V=v] < v

$S_v = [1-\rho^v]/[1-\rho]$, where $\rho = \lambda/(\lambda+\delta) < 1$

$S = E[\text{Sale price}] = (1-E[\rho^V])/[1-\rho] \leq E[V]$

$v = \varepsilon S_v$, or $S = E[V]/\varepsilon$, $\varepsilon > 1$

<u>If buyers are careful, then ε will be larger</u>

Imperial College London

Smart Price Formation

If the bidders select value distribution such as

$P(V=v) = \alpha^{v-1}(1-\alpha)$, **v>0**, $\alpha<1$,

Then E[V]$= 1/(1-\alpha)$, $E[\rho^V]=\rho(1-\alpha)/(1-\alpha\rho)$,

S= 1$/(1-\rho\alpha)$ **E[V] = S.**ε

$\varepsilon>1$: $\rho = \varepsilon(S - 1)/(\varepsilon S - 1) < 1$

Once ε and either E[V] or S are known,

the bidding rate or the decision rate, or their relationship, are set via $\rho=1/(1+\delta/\lambda)$

Networked Auctions

N Physically (Network) Interconnected Auctions for the same good (n(t),k(t))

- A client can only be at one of these auctions at a given instant of time
- $n(t) = (n_1(t), \ldots, n_N(t))$ where $n_i(t)$ is the number of bidders at auction i
- $k(t) = (k_1(t), \ldots, k_N(t))$ where $k_i(t)$ is the price currently attained at auction i
- $k_i(t) > 0$ implies that $n_i(t)>0$
- $n_i(t)=0$ implies that $k_i(t)=0$,
- While when $n_i(t)>0$ then $k_i(t)\geq 0$
- The Mobile Bidder Model – a bidder at auction i who does not have an outstanding (made but not yet accepted) bid may move from auction i to auction j with probability P(i,j) or leave the auction system with probability P(i,N+1)

Imperial College London

Networked Auctions

- $\gamma = (\gamma_1, \ldots, \gamma_N)$ where γ_i is the external arrival rate of bidders at auction i
- $\mu_i \geq 0$; the departure rate of bidders from auction i is:

  $\mu_i(y,x) = (y-1)\, \mu_i$, if $x>0$

  $\mu_i(y,0) = y\mu_i$

- The value of the good at auction i is the r.v. V_i with $\psi_I(x) = P[V_i > x]$ and

  $\psi_I(0) = 1$

- $\beta_i \geq 0$; the departure rate of bidders from auction i is:

  $\beta_i(y,x) = (y-1)\, \beta_I\, \psi_I(x)$, if $x>0$

  $\beta_i(y,0) = y\, \psi_I(0)$

- The rate at which bids are accepted at auction I is δ_i
- Rates are inverses of average values of exponentially distributed iid random variables

Imperial College London

Let us introduce the following notation. e_i will denote the $n - vector$ all of whose elements are zero, except for the $i - th$ element which is $+1$, and let $n_i^+ = n + e_i$, $n_i^- = n - e_i$ provided that $n_i > 0$, $n_{ij}^{+-} = n + e_i - e_j$ provided that $n_j > 0$, and $k_i^+ = k + e_i$ and $k_i^- = k - e_i$ provided that $k_i > 0$.

The Chapman-Kolmogorov (C-K) equations for the system are:

$$\frac{dp(n,k,t)}{dt} = \sum_{i=1}^{N} \{\ \gamma_i p(n_i^-, k, t) 1[n_i > 0] \tag{41}$$

$$+ \beta_i (n_i - 1)\psi_i(k_i - 1) p(n, k_i^-, t) 1[n_i > 0, k_i \geq 2]$$

$$+ \beta_i \psi_i(0) n_i p(n, k_i^-, t) 1[k_i = 1] \tag{42}$$

$$+ \mu_i P(i, D) p(n_i^+, k, t)(n_i 1[k_i > 0] + (n_i + 1) 1[k_i = 0])$$

$$+ \sum_{j=1}^{N} \mu_i n_i P(i,j) p(n_{ij}^{+-}, k, t) 1[n_j > 0, k_i > 0]$$

$$+ \sum_{j=1}^{N} \mu_i (n_i + 1) P(i,j) p(n_{ij}^{+-}, k, t) 1[k_i = 0]$$

$$+ \sum_{x=1}^{\infty} \delta_i p(n, k + x e_i) 1[k_i = 0] 1[n_i > 0]\ \}$$

$$- \sum_{i=1}^{N} \{\gamma_i + [(\delta_i + \mu_i(n_i - 1)) 1[n_i > 0, k_i > 0]$$

$$+ \beta_i \psi_i(k_i)(n_i - 1)] 1[n_i > 0, k_i > 0]$$

$$+ (\beta_i n_i + \mu_i n_i)[n_i > 0, k_i = 0]\ \} p(n, k, t)$$

Imperial College London

Equations for the Numbers of Bidders

$$\frac{dp(n,t)}{dt} = \sum_{i=1}^{N} \{\gamma_i p(n_i^-, t) 1[n_i > 0]$$

$$+ \mu_i n_i P(i, D) p(n_i^+, t)$$

$$+ \sum_{j=1}^{N} [\mu_i n_i P(i,j) p(n_{ij}^{+-}, t) 1[n_j > 0]$$

$$+ \sum_{i=1}^{N} \{ -(\gamma_i + \mu_i (n_i - 1) 1[n_i > 0]) p(n,t)$$

$$+ \mu_i P(i, D) p(n_i^+, k_i = 0, t)$$

$$+ \sum_{j=1}^{N} \mu_i n_i P(i,j) p(n_{ij}^{+-}, k_i = 0, t) 1[n_j > 0]$$

$$- \mu_i p(n, k_i = 0, t) 1[n_i > 0]\}$$

Analytical Solution for Active Bidders

Under the the *Active Bidders Assumption*, us suppose that that $p(n, k_i = 0, t) << p(n, t)$ for any n whose $i-th$ element $n_i > 0$. We are thus assuming that the probability that there are *no bids* when there are bidders at an auction is very small compared to the overall probability of the same state. The equations (48) then become:

$$\frac{dp(n,t)}{dt} \approx \sum_{i=1}^{N} \{ \gamma_i p(n_i^-, t)1[n_i > 0] \tag{49}$$

$$+ \mu_i n_i P(i, D)p(n_i^+, t) \tag{50}$$

$$+ \sum_{j=1}^{N} [\mu_i n_i P(i,j)p(n_{ij}^{+-}, t)1[n_j > 0]$$

$$- [\gamma_i + \mu_i(n_i - 1)]p(n,t)1[n_i > 0]\}$$

Result The stationary solution of equations (48) under the *Active Bidders Assumption* obtained by setting $\frac{dp(n,t)}{dt} = 0$ for all $n_i > 0$ is given by:

$$p(n) \approx \prod_{i=1}^{N} \frac{e^{-U_i}}{U_i} \frac{U_i^{n_i}}{(n_i - 1)!} \tag{51}$$

where $U_i = \frac{\Lambda_i}{\mu_i}$ and the Λ_i are the solution of the system of linear equations:

$$\Lambda_i = \gamma_i + \sum_{j=1}^{N} \Lambda_j P(j, i), \quad i = 1, \ldots, N \tag{52}$$

Active Bidders' Model
Given the Value of the Good

$$p(k|n) \approx \prod_{i=1}^{N} p_i(k_i|n_i), \text{ where} \quad (73)$$

$$p_i(k_i|n_i) = p_i(0|n_i) \prod_{l=1}^{k_i} [\frac{\beta_i \psi_{i,l-1}(n_i - 1)}{\gamma_i + \beta_i \psi_{i,l}(n_i - 1)} \quad (74)$$

and

$$p_i(0|n_i) = [1 + \sum_{k_i=1}^{\infty} \prod_{l=1}^{k_i} [\frac{\beta_i \psi_{i,l-1}(n_i - 1)}{\gamma_i + \beta_i \psi_{i,l}(n_i - 1)}]^{-1}. \quad (75)$$

where we have used the fact that the conditional probabilities sum to one.

Proof The proof is straightforward, and is based on substituting (74) in the equations (73). Note also that $p_i(0|0) = 1$.

Finally, we can write the steady-state solution under the ABA when bids and sales are frequent using (74) as:

$$p(n,k) \approx p(k|n)p(n) \quad (76)$$

$$\approx \prod_{i=1}^{N} \frac{e^{-U_i}}{U_i} \frac{U_i^{n_i}}{(n_i - 1)!} p(0|n_i) \prod_{l=1}^{k_i} [\frac{\beta_i \psi_{i,l-1}(n_i - 1)}{\gamma_i + \beta_i \psi_{i,l}(n_i - 1)} \quad (77)$$

Imperial College
London

Sensible Bidding Policies

i: The auction number
V_i: Item's value
S_{it} : Most recently observed selling price at auction I
$S_i \leftarrow S_{it}\alpha + S_i(1-\alpha)$: Historical average of selling price
P_i: current bid

R_i= D_i+G_i : effective time for reaching the seller
D_i: seller's decision time
G_i: CPN goal

SBP:

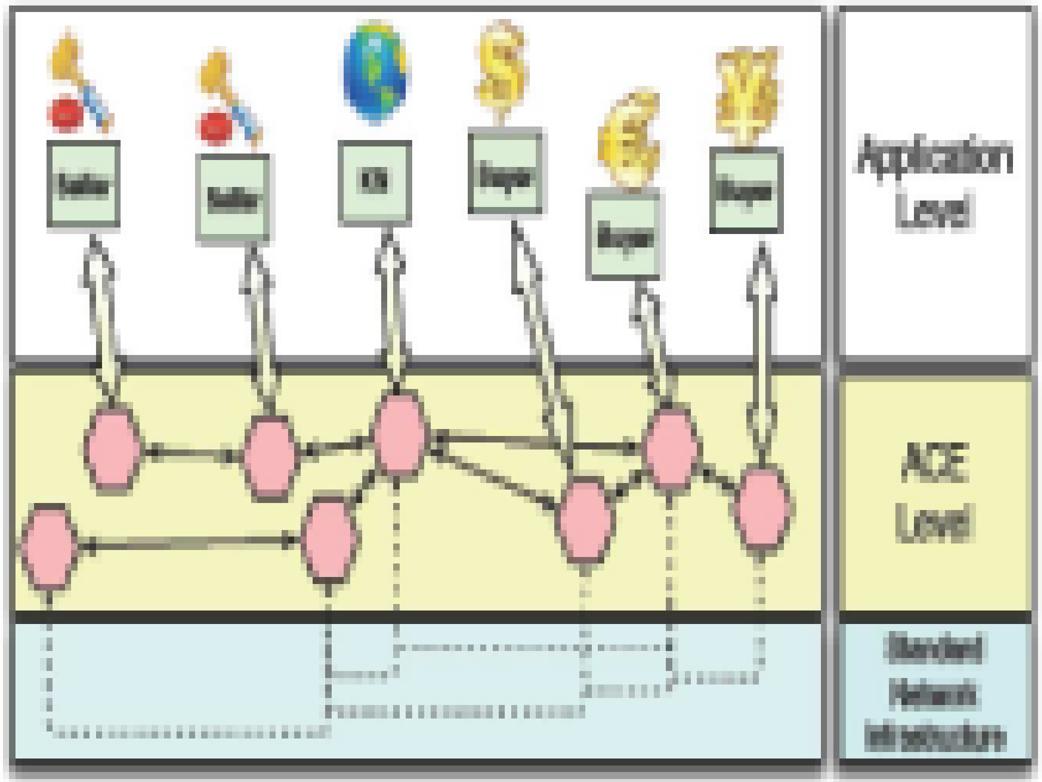
$$p_i = \frac{(V_i - S_i)^+ / R_i}{\sum_{i=1}^{n} (V_i - S_i)^+ / R_i}$$

SBP2:

$$p_i = \frac{(V_i - S_i)^+ * (V_i - P_i)^+ / R_i}{\sum_{i=1}^{n} (V_i - S_i)^+ * (V_i - P_i)^+ / R_i}$$

Imperial College
London

Implementation via Autonomic Agents
EU FP6
Cascadas



Imperial College London

Average Income $/sec

Imperial College London

Average Unsuccessful Bids

Imperial College London

Average Bids per/sec

Imperial College London

Average unsuccessful bids per/sec

Imperial College London

Conclusions

The World's Economy is a "Chain Reaction" of Electronic Transactions

Prices are Formed by feedback between markets and individual electronic age

Queuing network type models provide insight into price formation

Conclusions

G-Networks Provide Insight into a Wide Variety of Real Systems, such as:

- Communication Networks with Controls
- Load Balancing in Distributed Systems
- Communication in Neuronal Ensembles
- Communication in Gene Networks
- Chemical Reactions

They are also a new tool to study the Networked Economy

Imperial College London

Merci !

http://san.ee.ic.ac.uk

Imperial College
London