Time stealing: An adventure in tropical land

Marianne Johnson

Manchester, 16th November 2010

Time stealing: An adventure in tropical land

For Dave on the occasion of his 60th Birthday
(with apologies to Lewis Carroll)

Joint work with Dave Broomhead and Steve Furber.

Disclaimer

I'm an algebraist, not a computer scientist so I'll try not to say much about the application...

lest I get too far out of my depth!

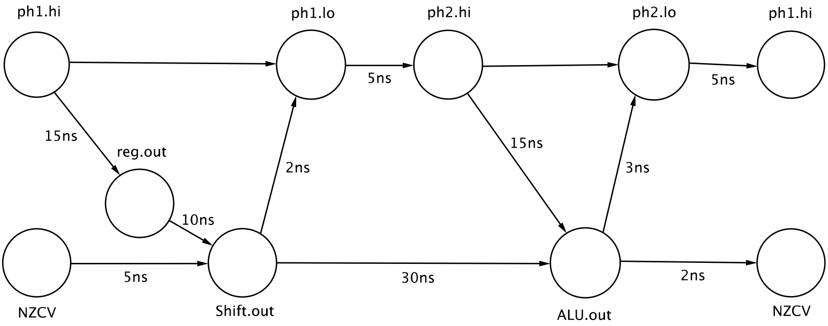
(As a special birthday treat for Dave, I'll divert any of your more difficult questions to him.)

Time stealing

'If everybody minded their own business' the Duchess said in a hoarse growl, 'the world would go around a deal faster than it does'.

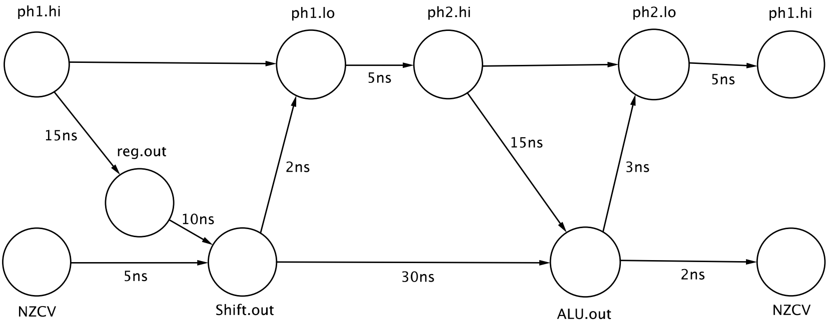
- **Time stealing** is a technique used to optimise the use of asynchronous processes in digital hardware.
- The system is clocked periodically using a multi-phase clock, but within each 'clock cycle' processes are allowed to interact **asynchronously**.
- Longer processes may be juxtaposed with shorter ones to allow the use of shorter clock periods than would otherwise be possible.

Motivating example



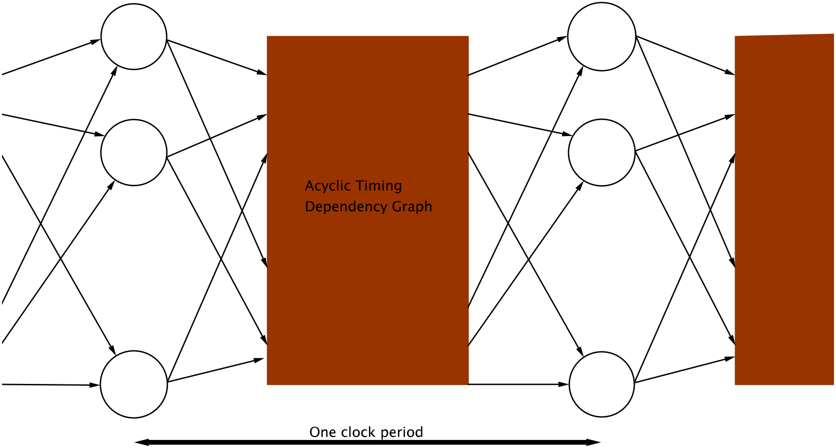
- ARM chips designed using transparent latches.
- The minimum clock period can be found by critical path analysis.
- Typically this involves looking for the longest logic path from a register output to a register input.
- Critical paths might therefore be several clock cycles long.

Timing dependency graph



- **Vertices** represent the time at which the input signal becomes valid.
- **Edges** represent causal links between signals (with delay due to linking logic).
- A kind of Hasse diagram - times are partially ordered.
- The timing dependency graph is a periodic structure.

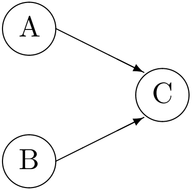
General picture



We are interested in Hasse diagrams with a periodic structure.

The algebra of timing

Imagine that a given process is waiting for inputs from two other processes.



The earliest time at which the input at C becomes valid is the **maximum** of the times at which the two input signals arrive. The input from A arrives at the time at which the input signal becomes valid at A **plus** the delay time from A to C.

$$t_C = \max(t_A + d_A, t_B + d_B).$$

Thus to study the dynamics of this problem we need to consider the operations of maximisation and addition on the real numbers.

Tropical land

The **tropical semiring** has elements $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$ and associative, commutative binary operations $\oplus$ and $\otimes$ defined by

$$a \oplus b = \max(a, b) \quad \text{and} \quad a \otimes b = a + b,$$

for all $a, b \in \mathbb{R}_{\max}$, where $\otimes$ distributes over $\oplus$.

The element $-\infty$ acts as a “zero” element, whilst the element 0 acts as a multiplicative identity. Thus for all $a \in \mathbb{R}_{\max}$:

$$\begin{aligned} a \oplus -\infty &= -\infty \oplus a = a, \\ a \otimes -\infty &= -\infty \otimes a = -\infty, \\ 0 \otimes a &= a \otimes 0 = a. \end{aligned}$$

We also have that for all $a \in \mathbb{R}_{\max}$ we have $a \oplus a = a$. We say that $\mathbb{R}_{\max}$ is an **idempotent semiring**.

On arithmetic in tropical land

'I'll try if I know all the things I used to know. Let me see: four times five is twelve, and four times six is thirteen, and four times seven is - oh dear! I shall never get to twenty at that rate!'

Of course, Alice should have known that four times five is nine, four times six is ten, and so forth.

With that in mind let us continue.

Tropical matrix algebra

We define matrices over $\mathbb{R}_{\max}$ in the usual way.
The operations $\oplus$ and $\otimes$ can then be generalised as follows:

$$\begin{aligned}(A \oplus B)_{i,j} &= A_{i,j} \oplus B_{i,j}, \text{ for all } A, B \in \mathbb{R}_{\max}^{m \times n} \\ (A \otimes B)_{i,j} &= \bigoplus_{k=1}^{l} A_{i,k} \otimes B_{k,j}, \text{ for all } A \in \mathbb{R}_{\max}^{m \times l}, B \in \mathbb{R}_{\max}^{l \times n}.\end{aligned}$$

Given an $n \times n$ matrix A we associate to it the weighted directed graph G_A as follows:

- G_A has n vertices labelled $1, \ldots, n$;
- If $A_{i,j} = -\infty$ then there is no edge from j to i;
- Otherwise, there is an edge from j to i labelled by $A_{i,j}$.

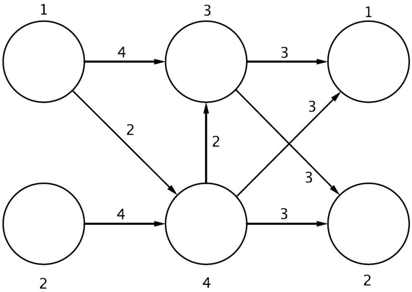
Eigenvalues - Curiouser and curiouser!

Given an $n \times n$ matrix A we look for a $\lambda \in \mathbb{R}_{\max}$ and a (non-trivial) vector $x \in \mathbb{R}^n_{\max}$ such that

$$A \otimes x = \lambda \otimes x.$$

Theorem If G_A is strongly connected then A possesses a unique eigenvalue. Moreover, the eigenvalue is the real number (i.e. not $-\infty$) equal to the maximal average delay of circuits in G_A.

A small example

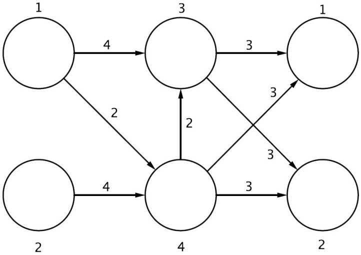


- If we identify vertices with the same label, the resulting graph is strongly connected.
- The matrix of delay times is

$$A = \left(\begin{array}{cc|cc} -\infty & -\infty & 3 & 3 \\ -\infty & -\infty & 3 & 3 \\ \hline 4 & -\infty & -\infty & 2 \\ 2 & 4 & -\infty & -\infty \end{array}\right)$$

- Thus A has unique eigenvalue $\lambda = 3.5...$

A small example (continued)

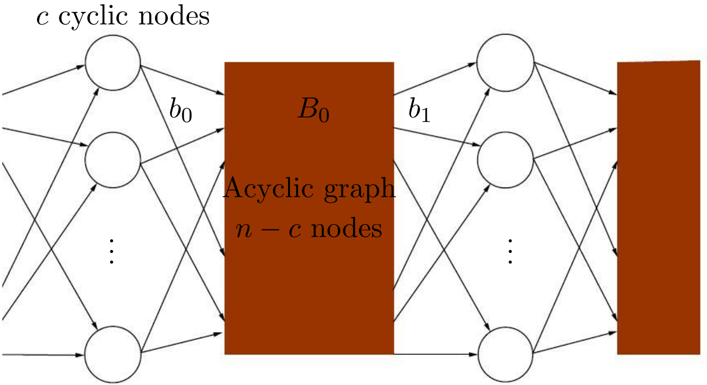


$$A = \left(\begin{array}{cc|cc} -\infty & -\infty & 3 & 3 \\ -\infty & -\infty & 3 & 3 \\ \hline 4 & -\infty & -\infty & 2 \\ 2 & 4 & -\infty & -\infty \end{array}\right)$$

Unique eigenvalue of A: $\lambda = 3.5$

- The eigenvalue represents the **mean delay** of the 'circuits' in the graph G_A (each node may be visited at most once).
- To calculate the minimum clock period, we need something a bit like this. However we don't really care how many internal nodes we pass through, or if we pass through these internal nodes more than once.
- We only care about how many times we pass through the cyclic nodes (and each cyclic node may be visited at most once).

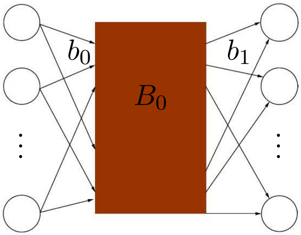
The general case



- Identify the 'cyclic nodes'.
- The matrix of delay times is

$$A = \left(\begin{array}{c|c} -\infty & b_1 \\ \hline b_0 & B_0 \end{array} \right)$$

Dynamics



$$A = \left(\begin{array}{c|c} -\infty & b_1 \\ \hline b_0 & B_0 \end{array}\right)$$

where b_0 is $(n-c) \times c$,
B_0 is $(n-c) \times (n-c)$,
b_1 is $c \times (n-c)$.

- For $i = 1, \ldots, n$ let $x_i(k)$ denote the time at which signal i becomes valid for the kth time.
- To get going we need an initial condition.
Suppose we know $x_1(1), \ldots, x_n(1)$.
- Let $A_0 = \left(\begin{array}{c|c} -\infty & -\infty \\ \hline b_0 & B_0 \end{array}\right)$ and $A_1 = \left(\begin{array}{c|c} -\infty & b_1 \\ \hline -\infty & -\infty \end{array}\right)$.
- Then $A = A_0 \oplus A_1$ and

$$x(k) = (A_0 \otimes x(k)) \oplus (A_1 \otimes x(k-1)).$$

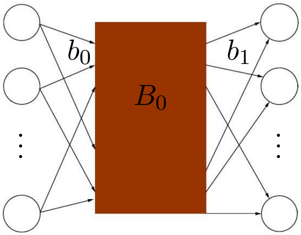
Strange tools: The Kleene star

Given an $n \times n$ matrix of delay times A, with acyclic timing dependency graph G_A, the Kleene star A^* is defined as

$$A^* = \bigoplus_{k \geqslant 0} A^{\otimes k}.$$

- $A^{\otimes k}_{i,j}$ gives the maximum delay of paths of length k in G_A from j to i.
- Since G_A is a finite acyclic graph, A^* is given by a sum of a finite number of terms.
- $A^*_{i,j}$ gives the maximum delay of paths in G_A from j to i.
- By substitution it is easy to check that $x = A^* \otimes b$ is a solution of $x = (A \otimes x) \oplus b$.

Dynamics



$$A = \left(\begin{array}{c|c} -\infty & b_1 \\ \hline b_0 & B_0 \end{array}\right), \; A = A_0 \oplus A_1 \text{ where}$$

$$A_0 = \left(\begin{array}{c|c} -\infty & -\infty \\ \hline b_0 & B_0 \end{array}\right), \; A_1 = \left(\begin{array}{c|c} -\infty & b_1 \\ \hline -\infty & -\infty \end{array}\right).$$

- Recall that $x_i(k)$ is the time at which signal i becomes valid for the kth time.
- Initial condition: $x_1(1), \ldots, x_c(1), x_{c+1}, \ldots, x_n(1)$.
- Then

$$x(k) = (A_0 \otimes x(k)) \oplus (A_1 \otimes x(k-1)).$$

- Thus, using the Kleene star, we find

$$x(k) = A_0^* \otimes A_1 \otimes x(k-1).$$

The form of our Kleene star

The dynamics of our system are governed by the system of equations

$$x(k) = A_0^* \otimes A_1 \otimes x(k-1).$$

Recall that the matrices A_0 and A_1 have a nice block matrix form, with lots of $-\infty$ entries. Using this nice block matrix structure it is then easy to check that:

$$\begin{aligned} A_0^* &= \left(\begin{array}{c|c} \mathrm{id} & -\infty \\ \hline B_0^* \otimes b_0 & B_0^* \end{array}\right), \\ A_0^* \otimes A_1 &= \left(\begin{array}{c|c} -\infty & b_1 \\ \hline -\infty & B_0^* \otimes b_0 \otimes b_1 \end{array}\right) \end{aligned}$$

So we only need to know $x_{c+1}(k-1), \ldots, x_n(k-1)$.

Eliminating the cyclic nodes

In other words, if we let $u(k) = (x_1(k), \ldots, x_c(k))^\top$ and $v(k) = (x_{c+1}(k), \ldots, x_n(k))^\top$, then

$$\begin{aligned} u(k) &= b_1 \otimes v(k-1) \\ v(k) &= B_0^* \otimes b_0 \otimes b_1 \otimes v(k-1). \end{aligned}$$

Finally, it can be shown that the minimum clock period is given by the eigenvalue of $B_0^* \otimes b_0 \otimes b_1$, and that this coincides with the heuristic given by the ARM designers.

Happy 60th Birthday Dave

...or more accurately,
happy 21,859th un-birthday!

Off with her head!