ILP-Based Synthesis for Sample Preparation Applications on Digital Microfluidic Biochips

Abhimanyu

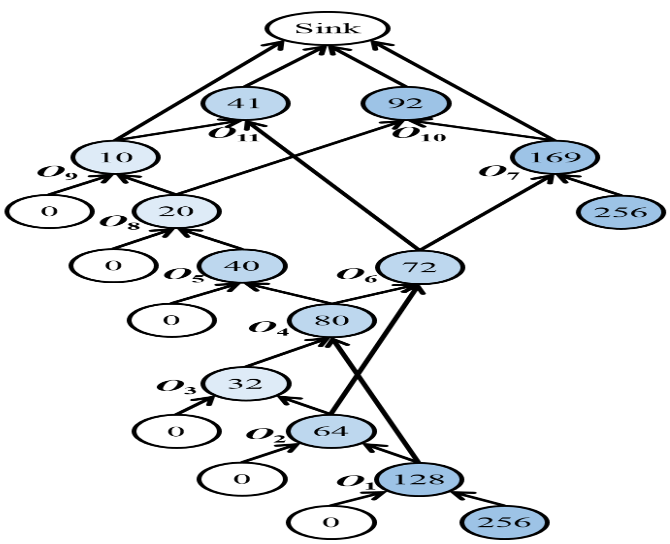
July 27, 2015

1 Introduction

This document describes a method to convert the information from a given application graph and chip architecture into a system of linear inequations which can be given as an input to the ILP solver to obtain the the binding and scheduling result.

2 Application Graph

The application graph gives the information of the required chronological order of the operations. To show how to build up the inequations from the application graph, it would be best to use a graph as an example.



Let O_i represent the operation MIX i.
Let s_i be the starting time step of operation O_i.
Let $T(O_i)$ be the time required to perform the operation O_i, it will be given in the Chip Architecture, but since only one type of operation i.e. MIX is being done, T(O_i) will be the same $\forall$ i.
Let n be the total number of mixing operations.

Operation	Inequation
O_1	$s_1 \geq 0$
O_2	$s_2 \geq s_1 + T(O_1)$
O_3	$s_3 \geq s_1 + T(O_1)$
O_4	$s_4 \geq s_2 + T(O_2)$
O_5	$s_5 \geq s_3 + T(O_3)$
O_6	$s_6 \geq s_4 + T(O_4)$
O_7	$s_7 \geq s_5 + T(O_5)$

We create a dummy variable s_8 such that $s_8 \geq s_6, s_8 \geq s_7$
***Objective*: Minimize** s_8

3 Scheduling result

Note: Time steps start from 0. T_{MAX} is the highest value the time step can take. We can define T_{MAX} to be the time it would take to finish the complete application doing only 1 operation at a time. It is just a very loose upper bound.

3.1 Mixing

We declare a class of binary variables $\{X_{it}\ \forall i,\ \forall t\}$ such that:

$$X_{it} = \begin{cases} 1 & \text{if } O_i \text{ is running at time step t} \\ 0 & \text{if } O_i \text{ is } \textit{not} \text{ running at time step t} \end{cases}$$

3.1.1 Formulation

For a given i and t:

if $(s_i \leq t < s_i + T(O_i))$
    $X_{it} = 1$
else
    $X_{it} = 0$

Inequations for the above if-else-condition.

$$\boxed{\begin{gathered} t - s_i \geq T_{MAX} * (X_{it} - 1) \\ s_i + T(O_i) - t > T_{MAX} * (X_{it} - 1) \\ \sum_{t=0}^{T_{MAX}} X_{it} = T(O_i) \end{gathered}}$$

3.2 Storage

Let E be the set of all edges of the graph which start and terminate on a MIX node and $\|E\| = n_e$.We declare a class of binary variables $\{Y_{et}\ \forall e,\ \forall t\}$ such that:

$$Y_{et} = \begin{cases} 1 & \text{if Droplet of egde e is being stored at time step t} \\ 0 & \text{if Droplet of egde e is not being stored at time step t} \end{cases}$$

3.2.1 Formulation

For a given e (where e is the edge between i and j) and t:

if $(s_i + T(O_i) \leq t < s_j)$
    $Y_{et} = 1$
else
    $Y_{et} = 0$

Inequations for the above if-else-condition.

$$\boxed{\begin{gathered} t - (s_i + T(O_i)) \geq T_{MAX} * (Y_{et} - 1) \\ s_j - t > T_{MAX} * (Y_{et} - 1) \\ \sum_{t=0}^{T_{MAX}} Y_{et} = s_j - (s_i + T(O_i)) \end{gathered}}$$

4 Chip Architecture Constraints

We have n_m modules available. In general, depending on the size of the module, it can be used either as mixer or a storage for n_r droplets. Here, we will be considering the the special case of $n_r = 1$. $\therefore$ we can have a total of n_m mixing and storage operations atmost.
We get the following constraint from this:

$$\boxed{\forall t \sum_{i=1}^{n} X_{it} + \sum_{e=1}^{E} Y_{et} \leq n_m}$$

where n is the number of mixing operations.

4.1 Binding result for Mixing

Let $\{M_{pi}\ \forall p, \forall i\}$ be a class of binary variables such that:

$$M_{pi} = \begin{cases} 1 & \text{if } O_i \text{ is bound to Mixer p} \\ 0 & \text{if } O_i \text{ is } \textit{not} \text{ bound to Mixer p} \end{cases}$$

- To ensure that an operation remains bound to the same module throughout the time it is being executed, we have the following inequation:

  $$\boxed{\sum_{p=1}^{n_m} M_{pi} = 1}$$

- If two operations i, j are bound to the same module p, then there can't be a time step t when both of them are running simultaneously.

  For all mixing operations i, j and $i \neq j$
  if($M_{pi} = M_{pj} = 1$)
  $\forall t\ X_{it} + X_{jt} < 2$

  The in-equations corresponding to the above are as follows:

  $$\boxed{\begin{array}{c} X_{it} + X_{jt} \geq M_{pi} + M_{pj} - 2 \\ X_{it} + X_{jt} \leq 3 - (M_{pi} + M_{pj}) \end{array}}$$

4.2 Binding result for Storage

Let $\{R_{pe}\ \forall p, \forall e\}$ be a class of binary variables such that:

$$R_{pe} = \begin{cases} 1 & \text{if droplet of edge e is bound to Mixer p} \\ 0 & \text{if droplet of edge e is } \textit{not} \text{ bound to Mixer p} \end{cases}$$

- To ensure that a droplet remains bound to the same module throughout the time it is being executed, we have the following inequation:

  $$\boxed{\forall e \sum_{p=1}^{n_m} R_{pe} \leq 1}$$

- If two edges i, j are bound to the same module p, then there can't be a time step t when both of them are being stored.

  For all edges i, j and $i \neq j$
  if($R_{pi} = R_{pj} = 1$)
  $\forall t\ Y_{it} + Y_{jt} < 2$

The inequations corresponding to the above are as follows:

$$\boxed{\begin{array}{c} Y_{it} + Y_{jt} \geq R_{pi} + R_{pj} - 2 \\ Y_{it} + Y_{jt} \leq 3 - (R_{pi} + R_{pj}) \end{array}}$$

4.3 Additional Constraint

We also need to ensure that a module should not be used as both a reservoir and mixer at the same time. We have the following in-equation for this.

$$\boxed{\forall p, \forall t \sum_{i=1}^{n} M_{pi} * X_{it} + \sum_{e=1}^{E} R_{pe} * Y_{et} < 2}$$

Interpretation For a given module p, the first summation is 1 at the time step when an operation bound to this module is running at that time step. The in-equation prevents the module from storing any droplet at that time. Similarly, the second summation is 1 at the time step when a droplet bound to this module is being stored. The in-equation prevents the module from any mixing operation.
We can use the following methodology for AND formulation.

Inequations for $L_{pit} = M_{pi} * X_{it}$
$L_{pit} \geq M_{pi} + X_{it} - 1$
$L_{pit} \leq M_{pi} + X_{it}$
$L_{pit} \leq 1 - (M_{pi} - X_{it})$
$L_{pit} \leq 1 - (X_{it} - M_{pi})$

Inequations for $G_{pet} = R_{pe} * Y_{et}$
$G_{pet} \geq R_{pe} + Y_{et} - 1$
$G_{pet} \leq R_{pe} + Y_{et}$
$G_{pet} \leq 1 - (R_{pe} - Y_{et})$
$G_{pet} \leq 1 - (Y_{et} - R_{pe})$