Tightening worst-case timing analysis of Tilera-like NoC architecture

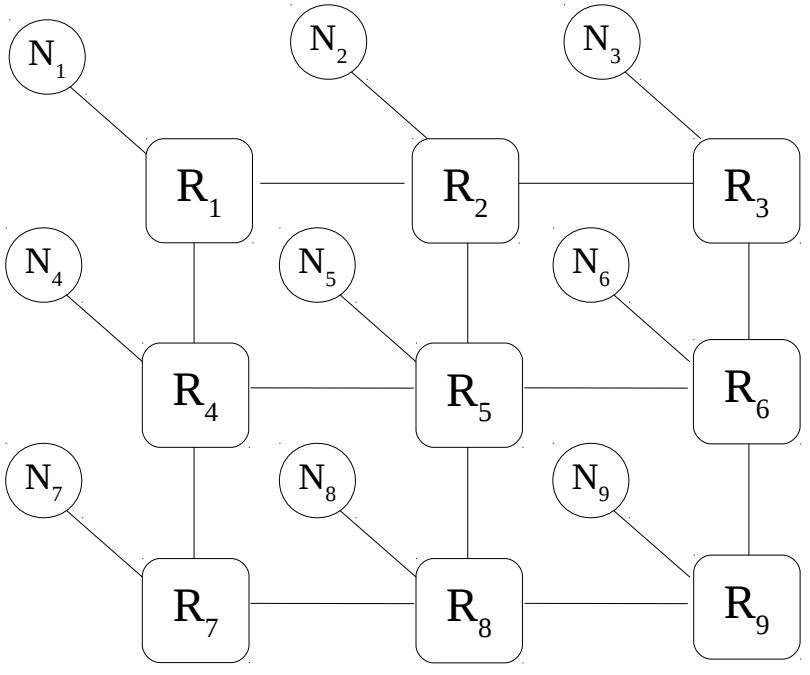
Hamdi Ayed, Jérôme Ermont, Jean-luc Scharbarg, Christian Fraboul

University of Toulouse

18th Euromicro Conference on Real-Time Systems
ECRTS 2016

WiP session

Network-on-Chip (NoC)



N_i Processing Element /Memory Controller /Off-chip interface

R_i Router

Network-on-Chip (NoC)

- NoC for **real-time** applications
  - → Bounded network latency
  - → Worst-case traversal time (WCTT) analysis

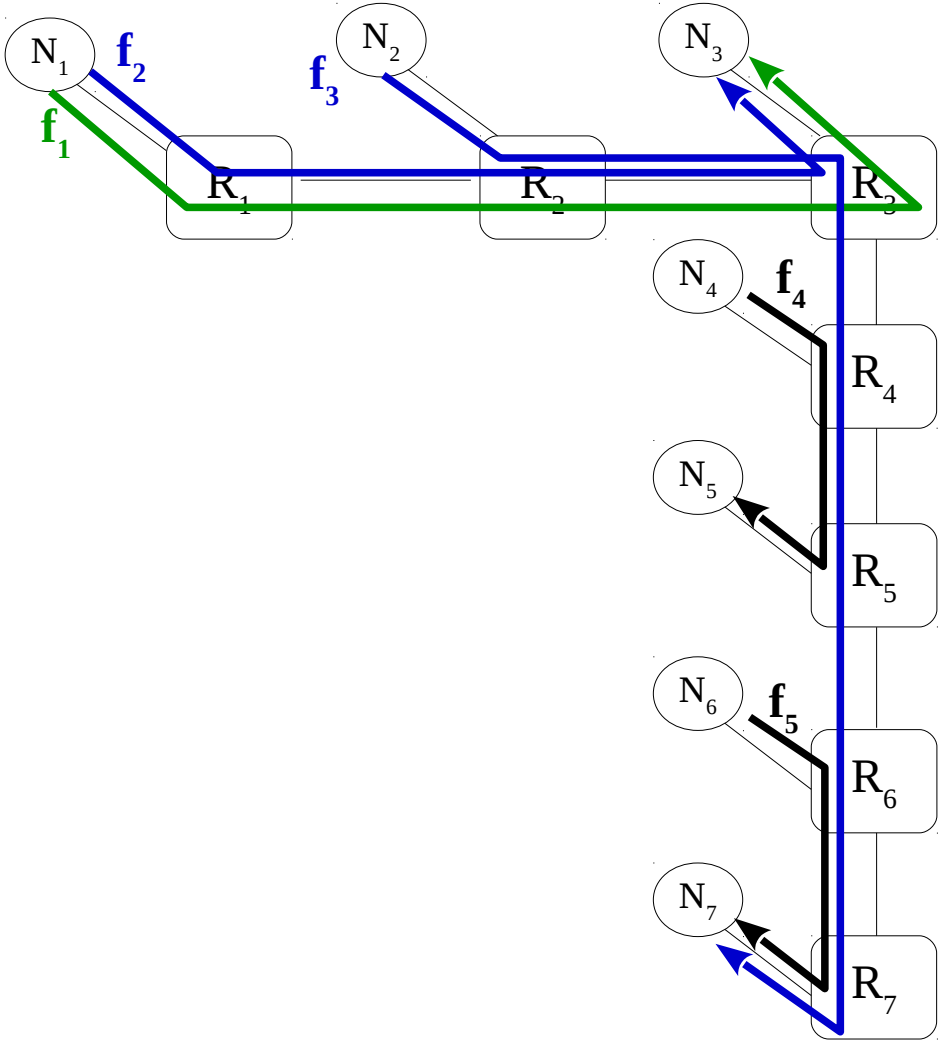
Network-on-Chip (NoC)

- NoC for **real-time** applications
  - → Bounded network latency
  - → Worst-case traversal time (WCTT) analysis
- Timing analysis techniques
  - Network calculus (NC)
  - **Recursive calculus** (RC)

Network-on-Chip (NoC)

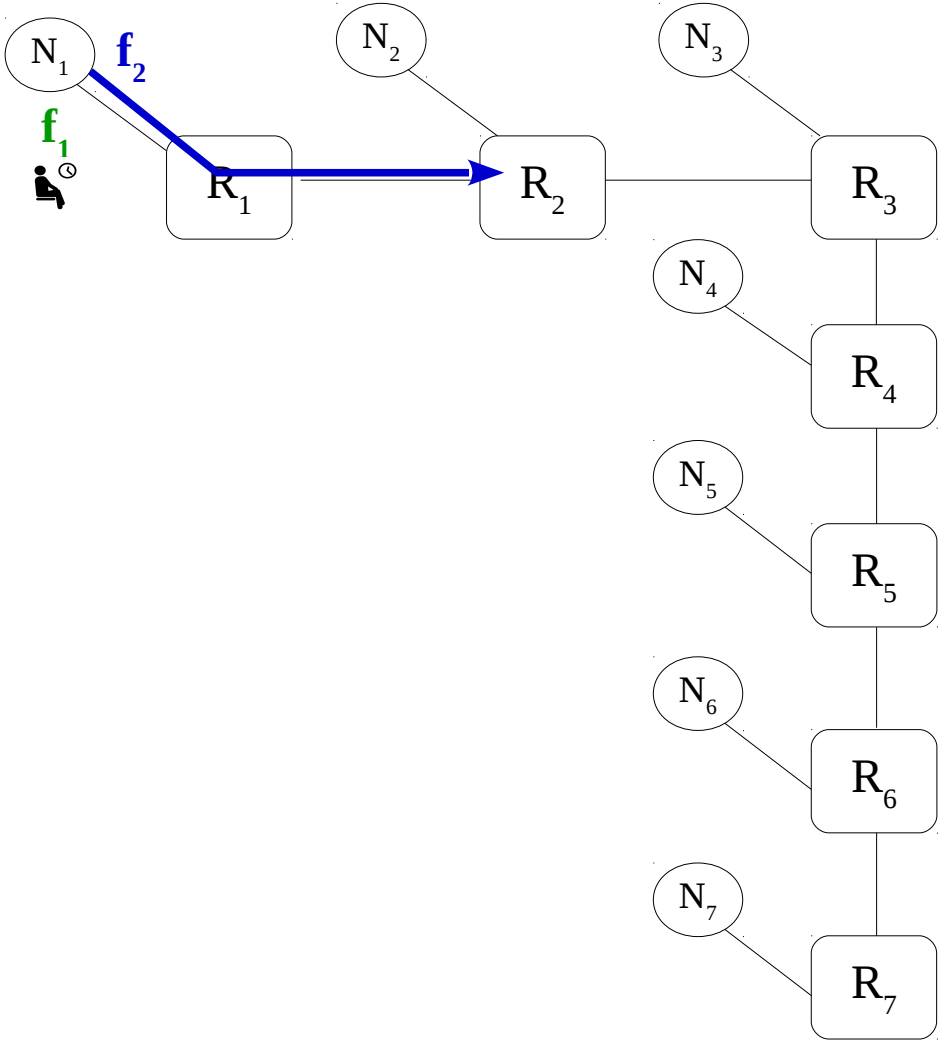
- NoC for **real-time** applications
  - → Bounded network latency
  - → Worst-case traversal time (WCTT) analysis
- Timing analysis techniques
  - Network calculus (NC)
  - **Recursive calculus** (RC)
- Our study
  - Tilera TILE64-like NoC
  - Improve WCTT bounds of Recursive Calculus analysis

Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

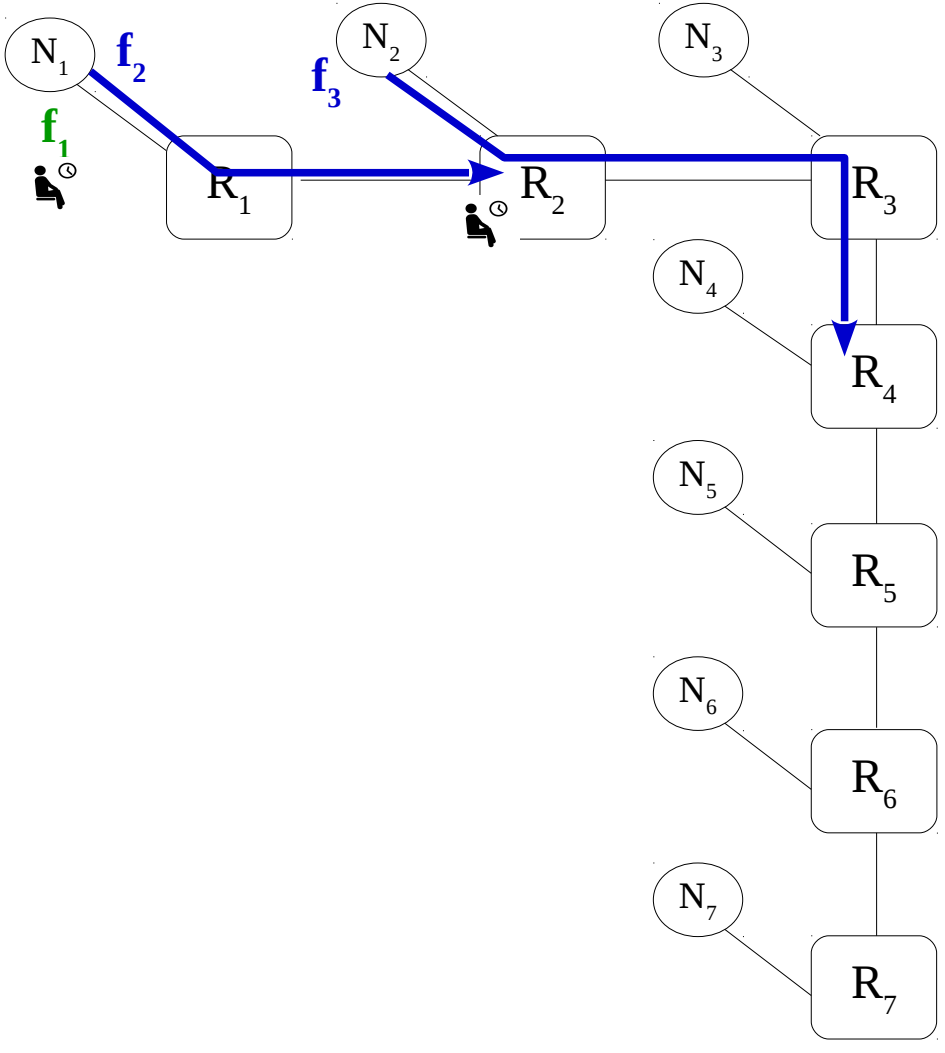
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

  → $\{f_2, \mathbf{f_1}\}$

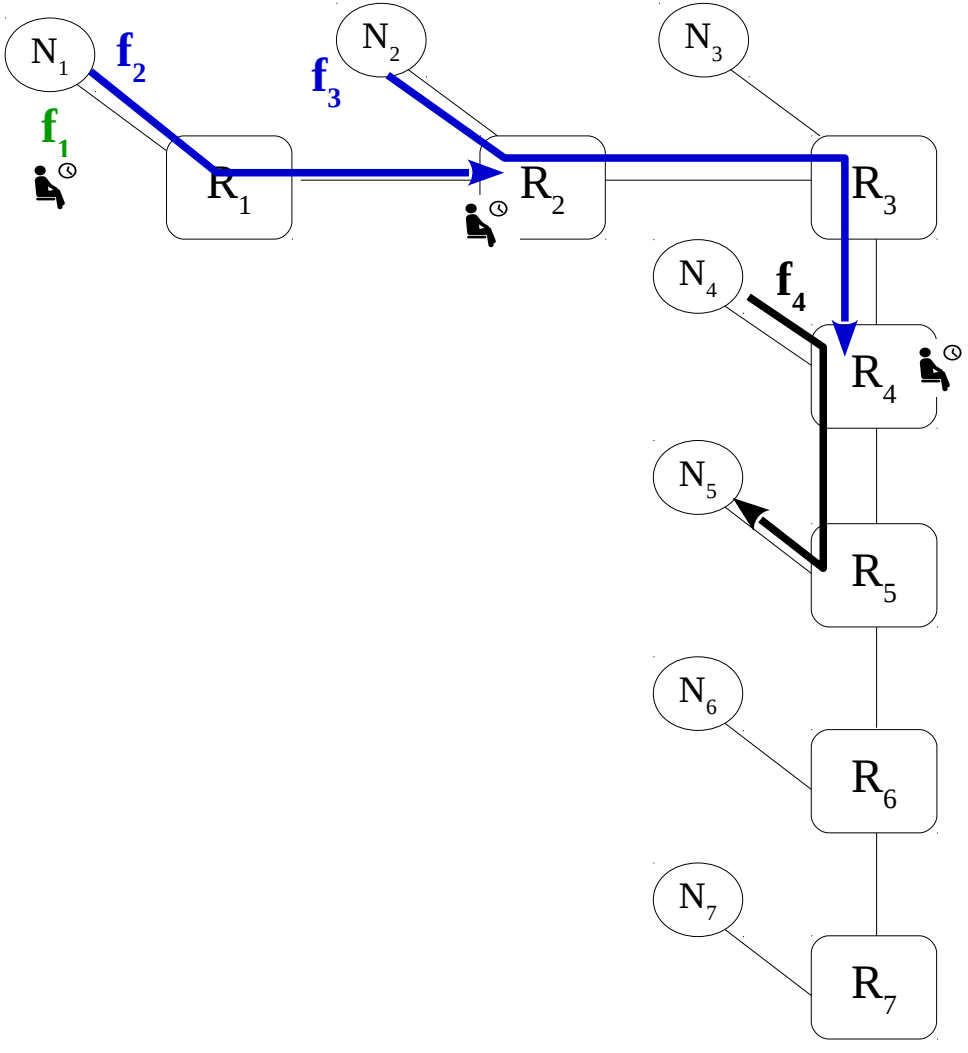
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_3, f_2, \mathbf{f_1}\}$

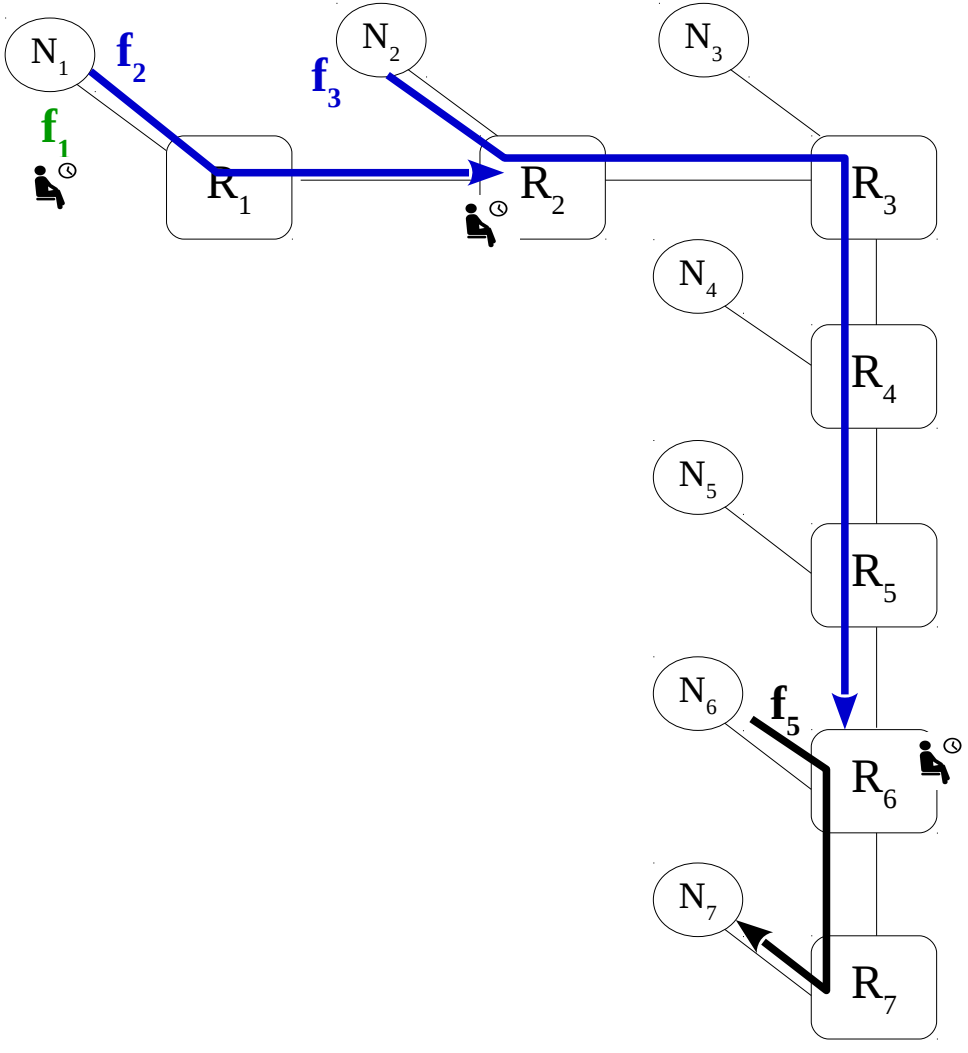
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_4, f_3, f_2, \mathbf{f_1}\}$

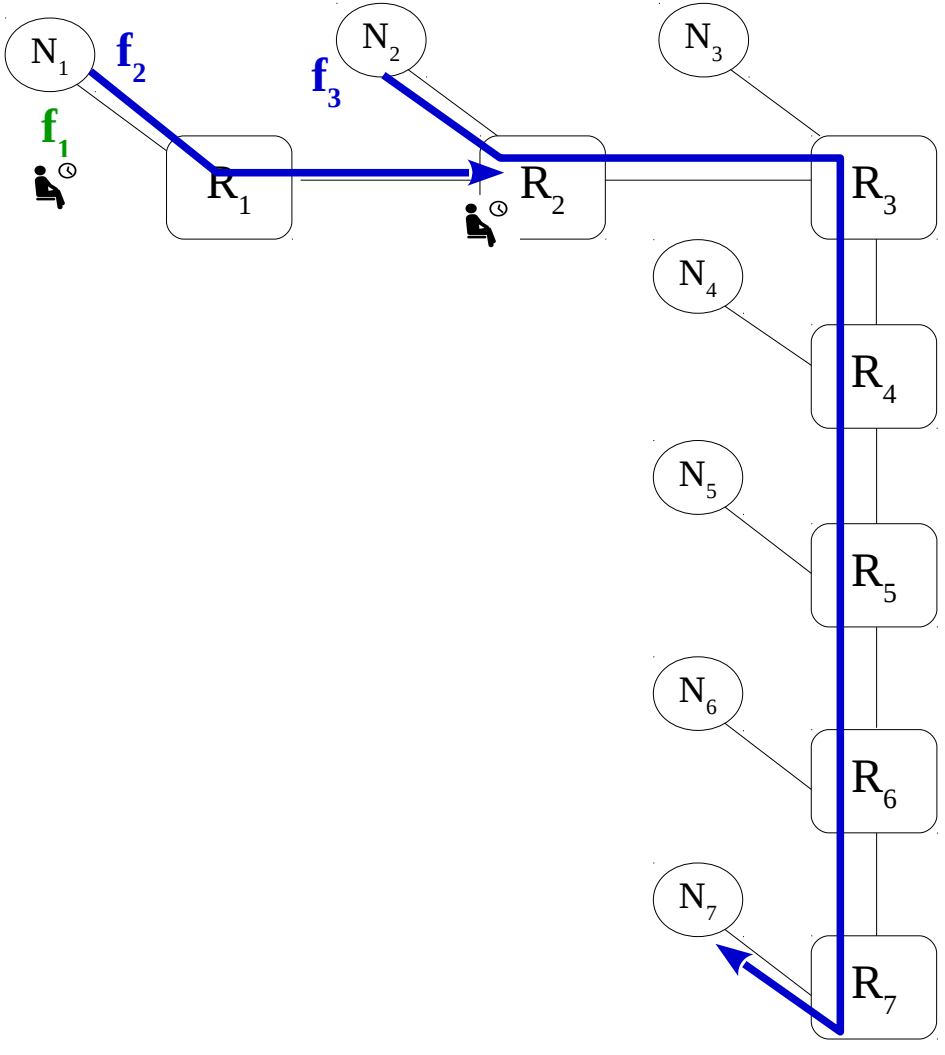
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, \mathbf{f_1}\}$

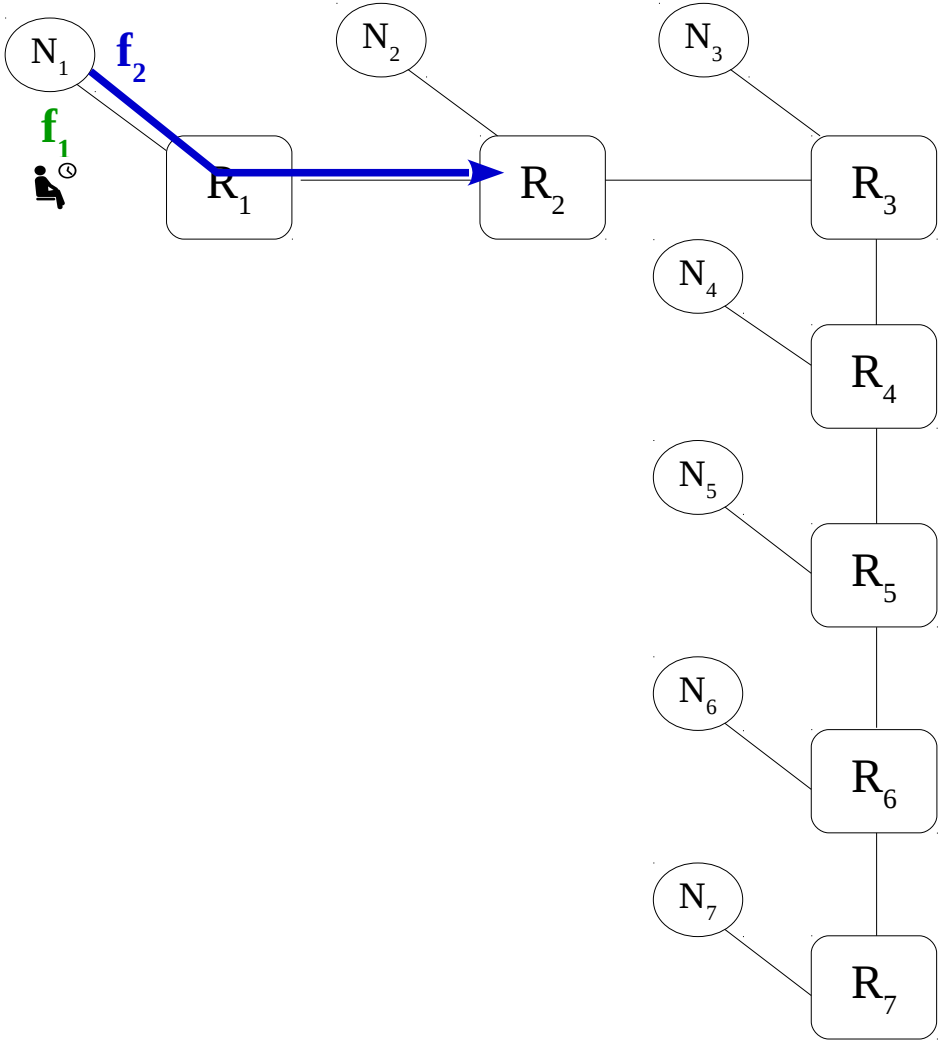
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, \mathbf{f_1}\}$

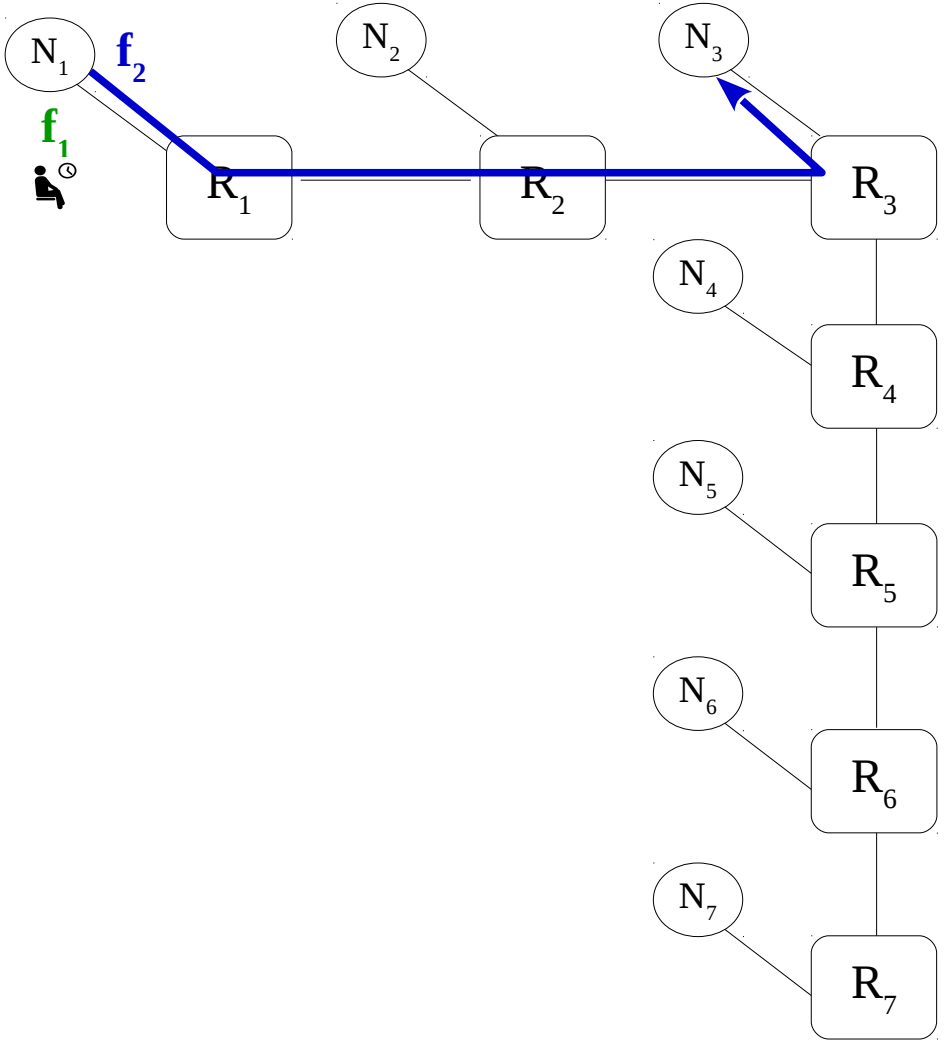
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, \mathbf{f_1}\}$

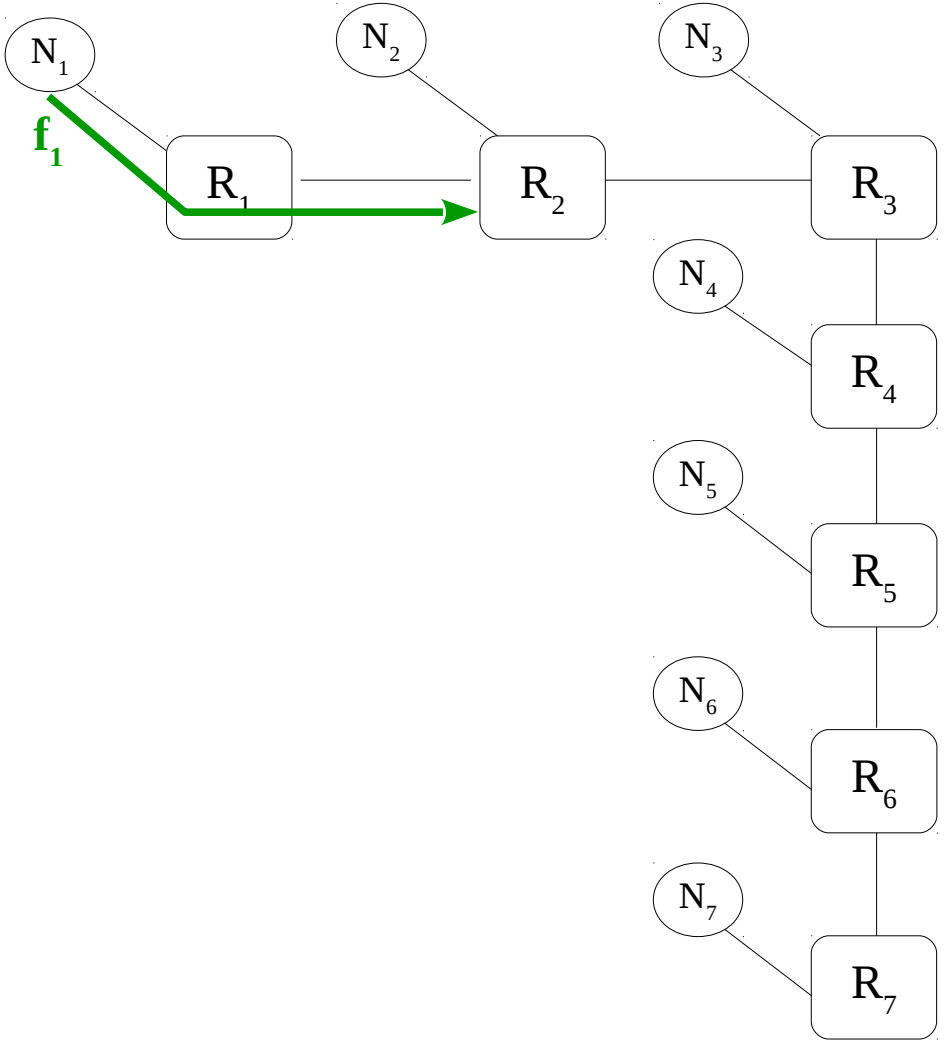
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, \mathbf{f_1}\}$

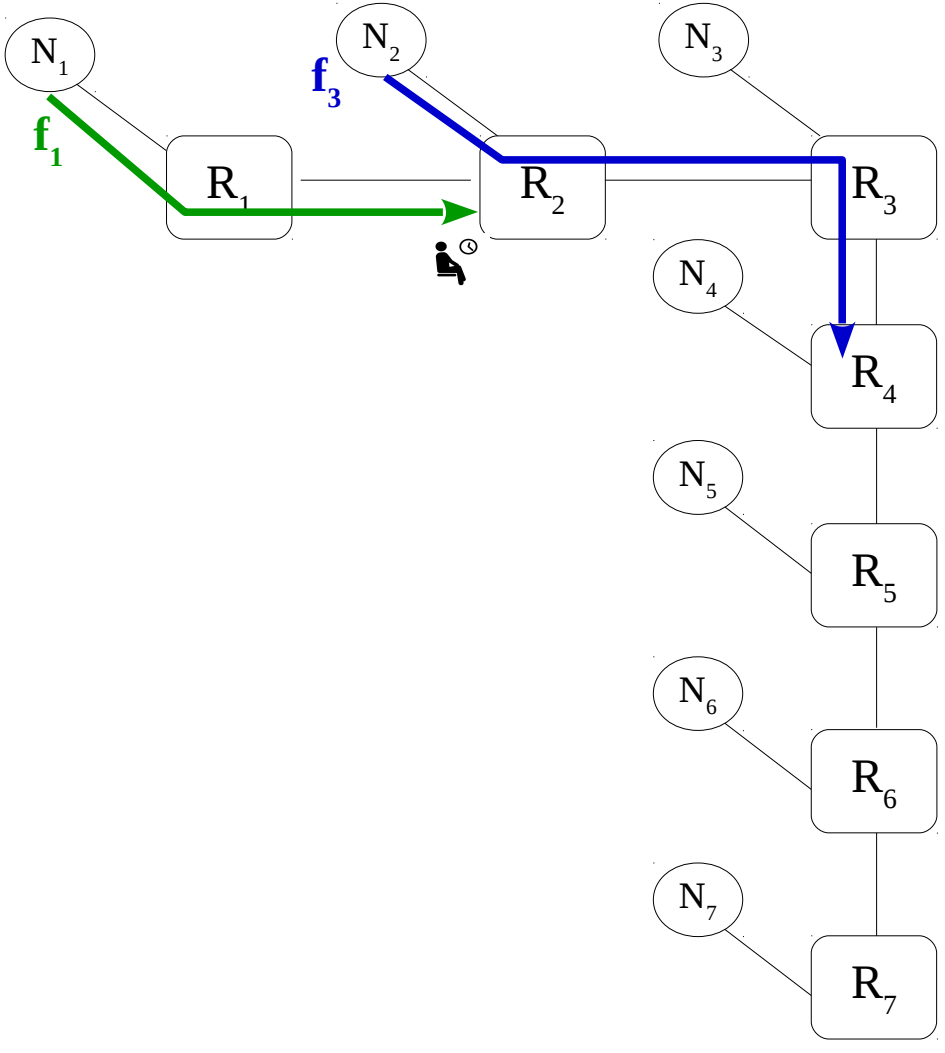
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, \mathbf{f_1}\}$

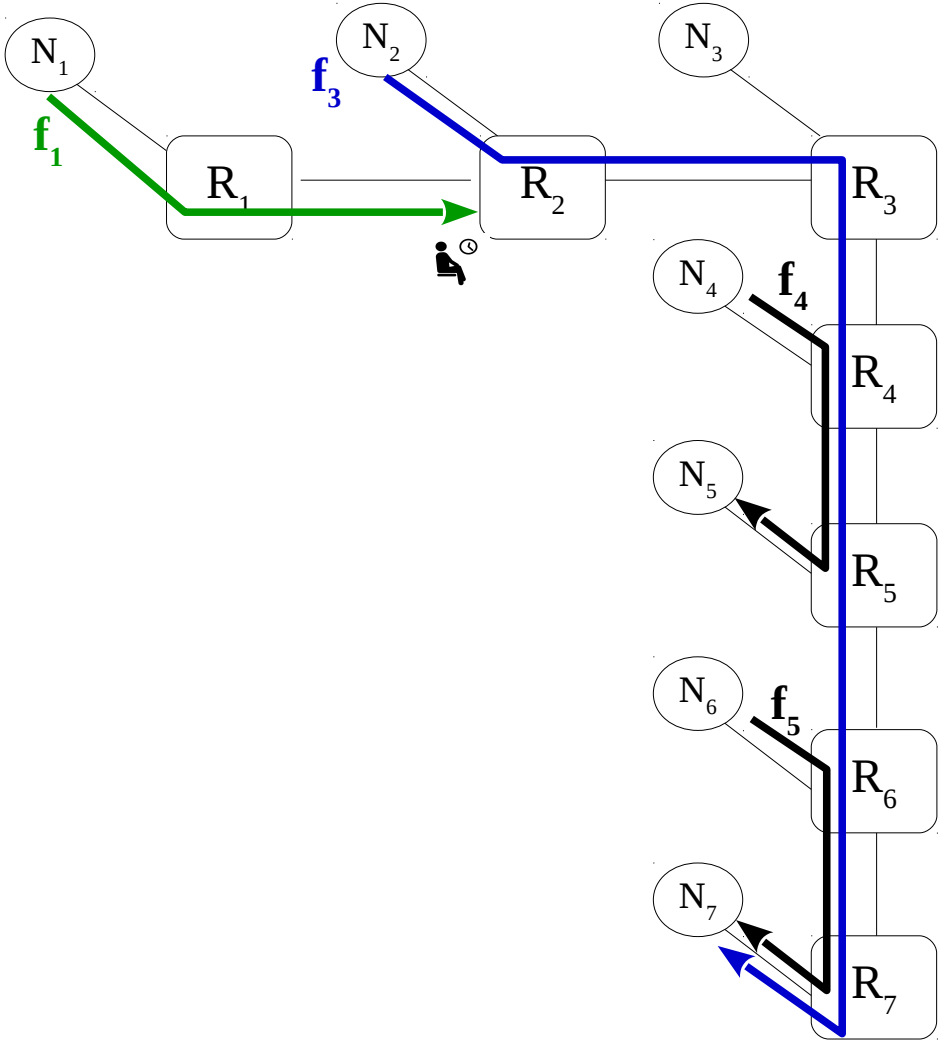
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, f_3, \mathbf{f_1}\}$

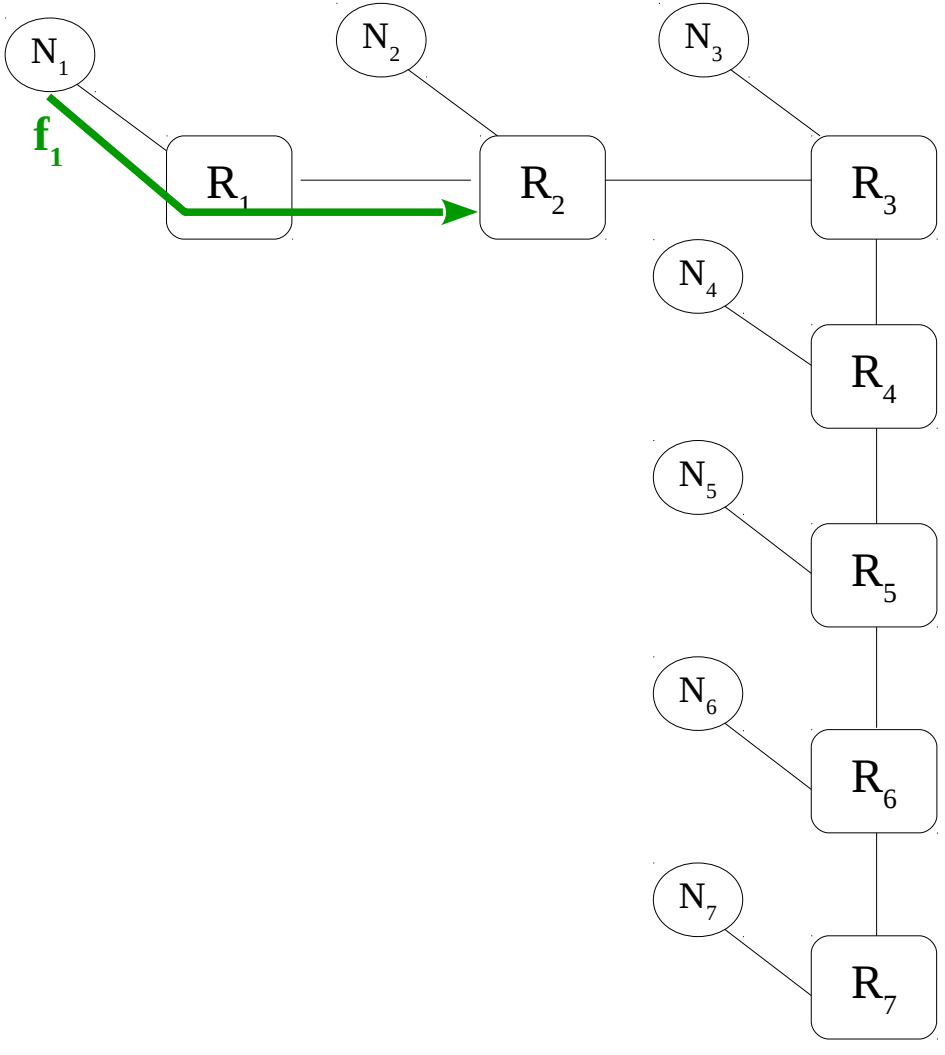
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, f_5, f_4, f_3, \mathbf{f_1}\}$

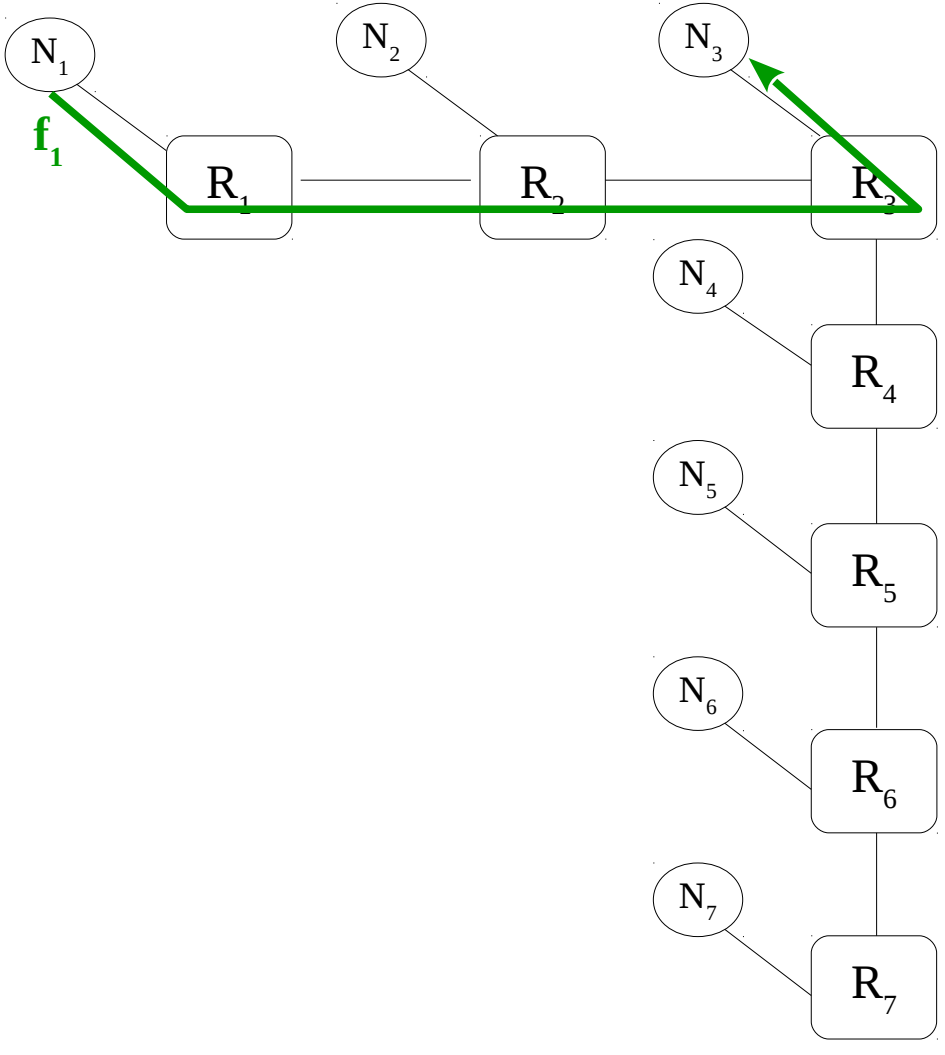
Existing recursive timing analysis



- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

    → $\{f_5, f_4, f_3, f_2, f_5, f_4, f_3, \mathbf{f_1}\}$

Existing recursive timing analysis



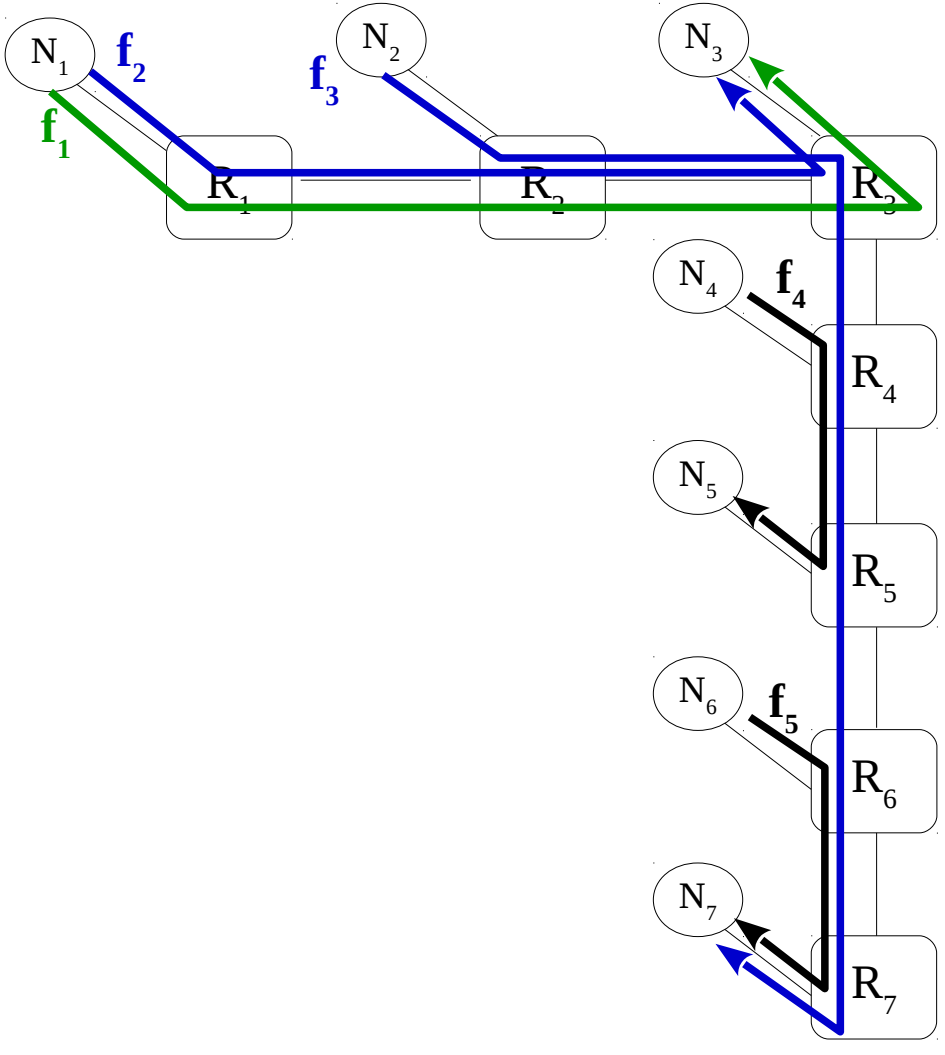
- Wormhole routing
- Round robin arbitration
- Flow level analysis
  - Ignores available buffer capacity
  - Maximal flow rate assumption

  → $\{f_5, f_4, f_3, f_2, f_5, f_4, f_3, \mathbf{f_1}\}$

  - Packet size = 4 flits
  - Transmission rate = 1 flit/cycle

  → WCTT(f_1) = 55 cycles

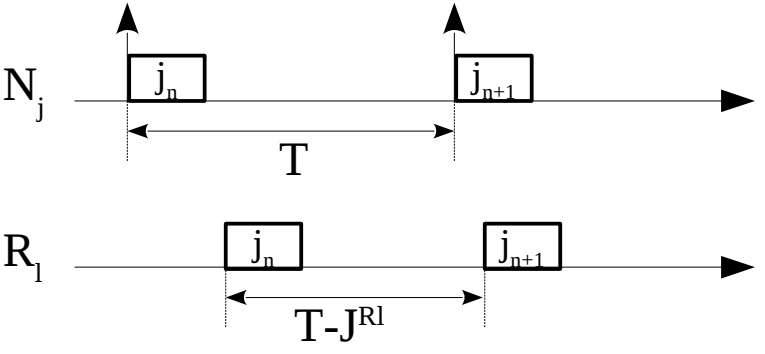
Enhanced approach



- Buffer effect
  - Buffer size = 3 flits

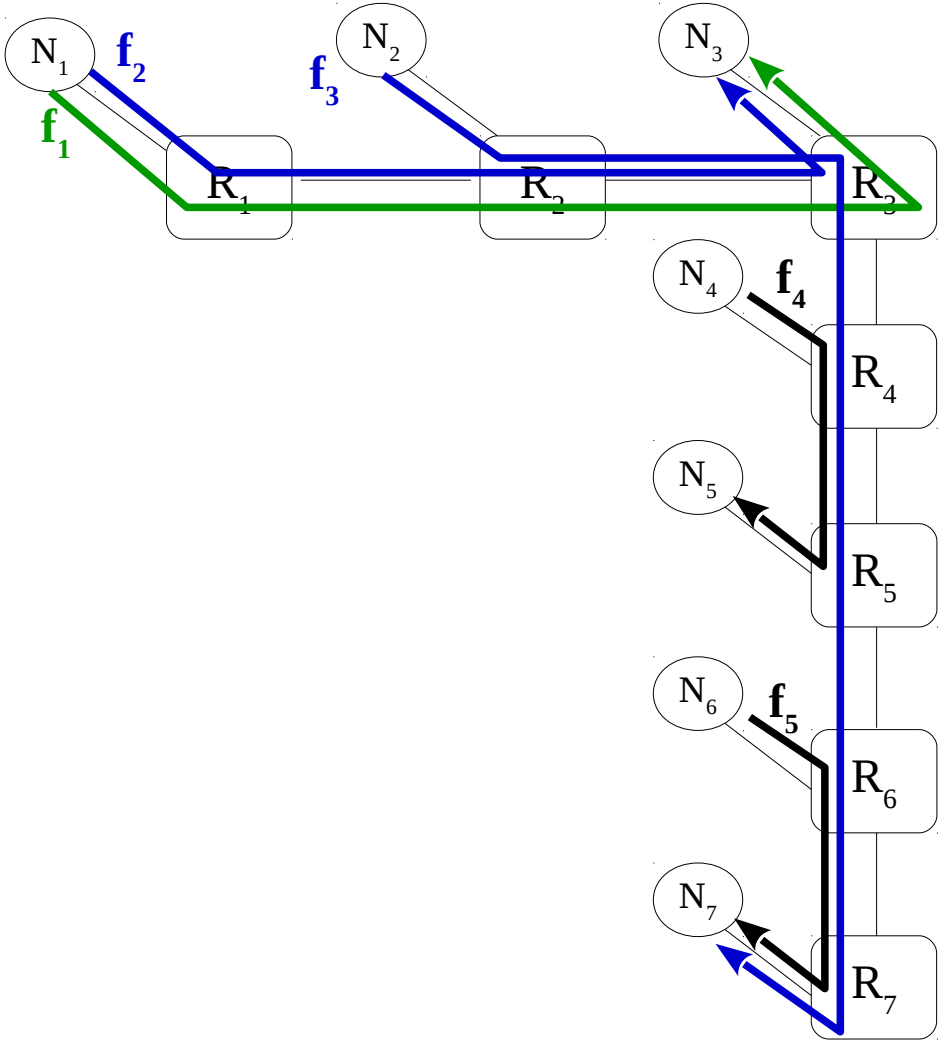
    → f_5 has no impact on f_1
- Minimim inter-release time

  → bounded number of packets released during [a,a+t]



$$Nb(f_j, R_l, t) = \left\lceil \frac{t}{T_j - J_j^{R_l}} \right\rceil \rightarrow Nb(f_3, R_2, WCTT^0(f)) = 1$$

Enhanced approach



- Buffer effect

  {~~f_5~~, f_4, f_3, f_2, ~~f_5~~, f_4, f_3, $\mathbf{f_1}$}

- Minimim inter-release time

  {f_5, f_4, f_3, f_2, ~~f_5~~, ~~f_4~~, ~~f_3~~, $\mathbf{f_1}$}

  or

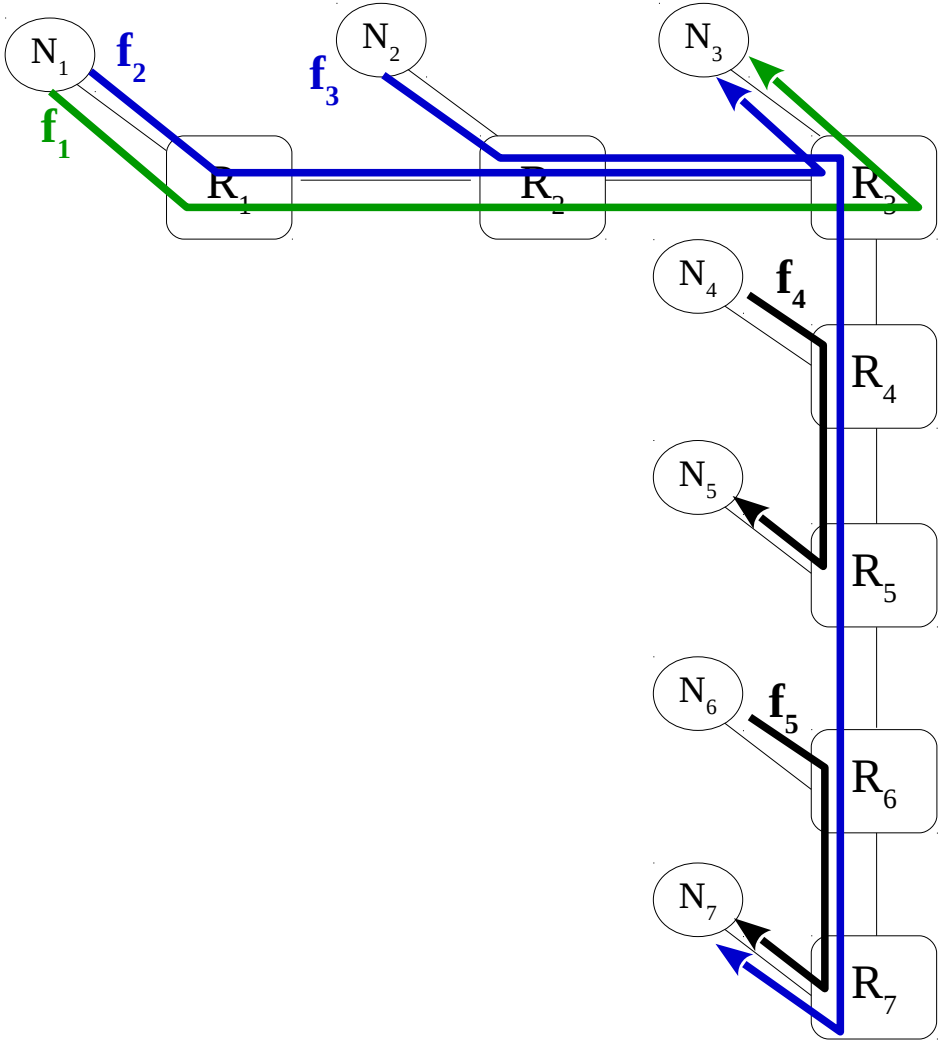
  {~~f_5~~, ~~f_4~~, ~~f_3~~, f_2, f_5, f_4, f_3, $\mathbf{f_1}$}

  or

  {f_5, f_4, f_3, f_2, ~~f_5~~, ~~f_4~~, f_3, $\mathbf{f_1}$}

  or …

Enhanced approach



- Combined effect

  $\{\cancel{f_5}, f_4, f_3, f_2, \cancel{f_5}, \cancel{f_4}, \cancel{f_3}, \mathbf{f_1}\}$

  or

  $\{\cancel{f_5}, \cancel{f_4}, \cancel{f_3}, f_2, \cancel{f_5}, f_4, f_3, \mathbf{f_1}\}$

  $\rightarrow \quad WCTT^{1}(f_1) = 24\,cycles$

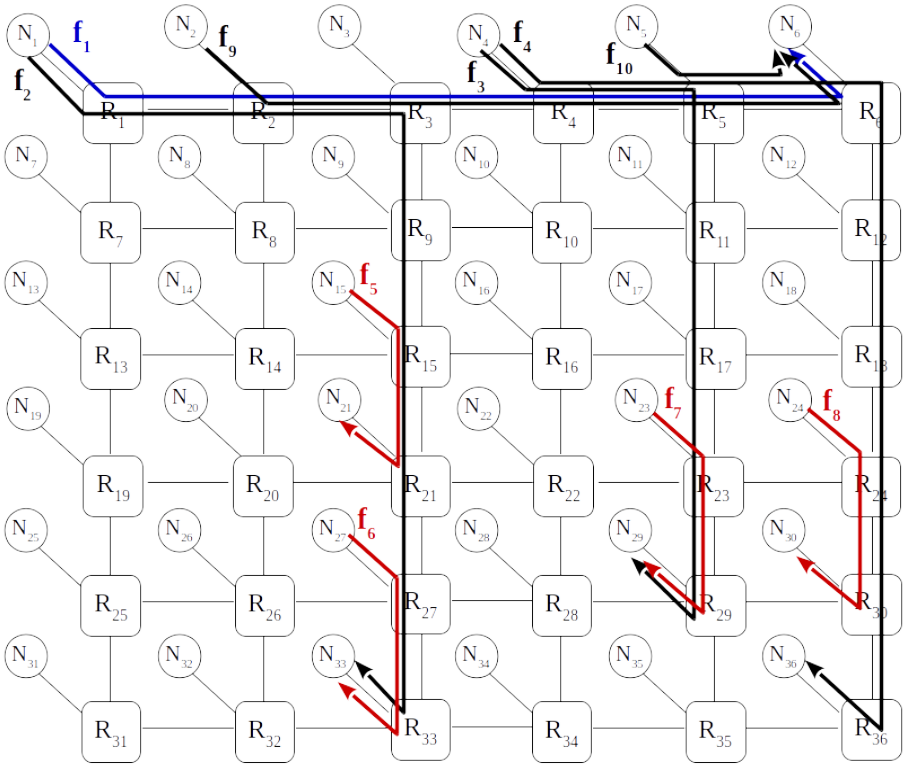
- Iteration to convergence
  - Stop when $WCTT^{n+1}(f_1) = WCTT^{n}(f_1)$

Preliminary results

- Real-time application

flow	Period (cycles)	Packet size (flits)
f_i	500	10

flow	Initial RC bounds (cycles)	Buffer effect (cycles)	Period effect (cycles)	Combined effect (cycles)
f1	304	251	121	111
f2	304	264	121	98
f3	143	112	86	78
f4	143	127	86	79
f5	44	40	27	25
f6	24	24	23	22
f7	27	25	25	23
f8	24	24	23	22
f9	167	145	89	82
f10	50	44	30	27



→ Up to 65 % WCTT bound reduction