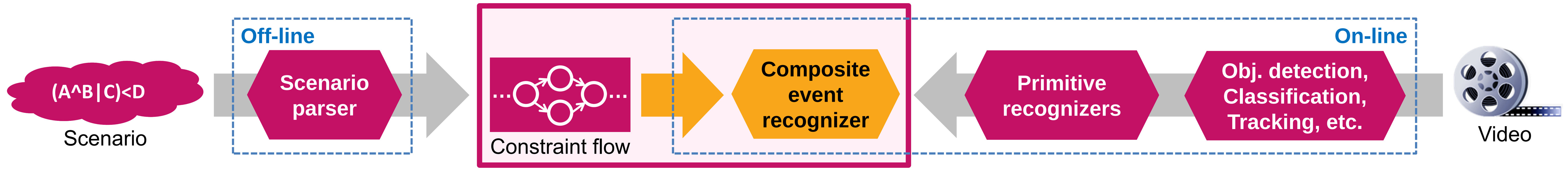
Scenario-Based Video Event Recognition by Constraint Flow

Suha Kwak   Bohyung Han   Joon Hee Han

{mercury3, bhhan, joonhan}@postech.ac.kr, *POSTECH*, Korea



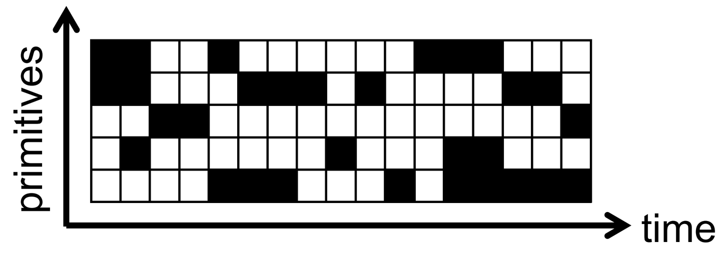
Introduction

- ***Scenario-based video event recognition***

  Recognizing events in videos with given knowledge (*scenario*) about how they happen.
- ***Contributions***
  - An easy and flexible scenario description method
  - Online and exact inference of events without heuristics

Goal & Challenge

- ***Search for the best video interpretation***



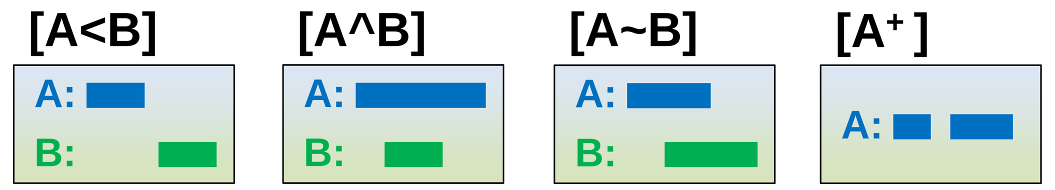
A video interpretation = a binary matrix indicating occurrences of primitives over time

- The number of interpretations increases exponentially over time → which one is feasible and optimal?
- We search the best interpretation by dynamic programming with constraint flow.

Scenario Description Method

A *scenario* = a set of temporal-logical *constraints for the arrangements of time intervals of primitives*

- ***Temporal relationships* (<, ^, ~, +)**



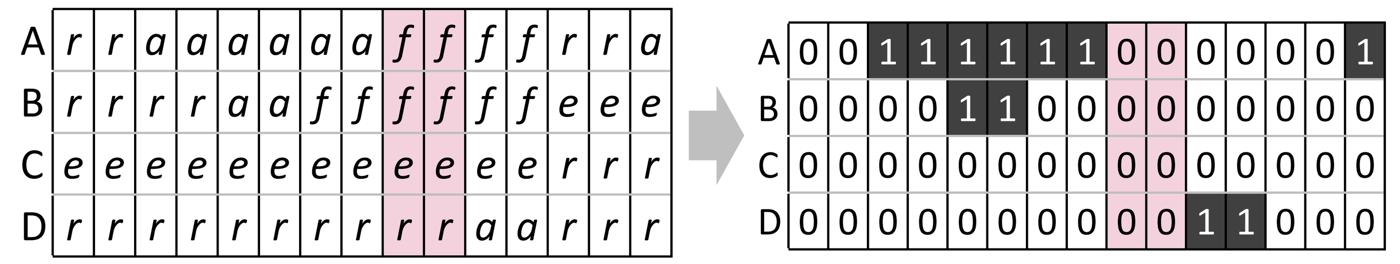
- ***Logical relationships* (&, |)**
  - **[A&B]**: both of A and B occur (no temporal ordering).
  - **[A | B]**: only one of A and B occurs.
- ***Dummy element* (#)**
- ***Parentheses***
  - Precedence (parentheses > logical > temporal)
  - Hierarchical description

Constraint Flow

- ***A dynamic configuration of scenario constraints***
  - Transitions (edges) among combinatorial states (vertices)
  - Feasible interpretations must follow the constraint flow!
- ***Automatic construction of constraint flow***
  - Breadth-first ***search*** and ***generation*** of flow vertices

Recognition with Constraint Flow

- ***Flow tracing to generate feasible interpretations***



A flow trace, $v_{1:15}$ (from v^1 to v^4)   A feasible interpretation, $x(v_{1:15})$

- ***Score function for traces***
  - Observation agreement & Penalty for idle intervals

$$f(v_{1:t}) = p(x(v_{1:t})|O_{1:t}) \cdot \mathcal{B}(v_{1:t})$$

$$\propto p(O_t|x(v_t)) \cdot \exp(-\beta I_b(v_t)) \cdot f(v_{1:t-1})$$

- ***Online, exact inference on a bounded search space***
  - Optimization by *dynamic programming*

$$\max_{v_{1:t-1}} f([v_{1:t-1}, v_t = v^k]) \propto$$

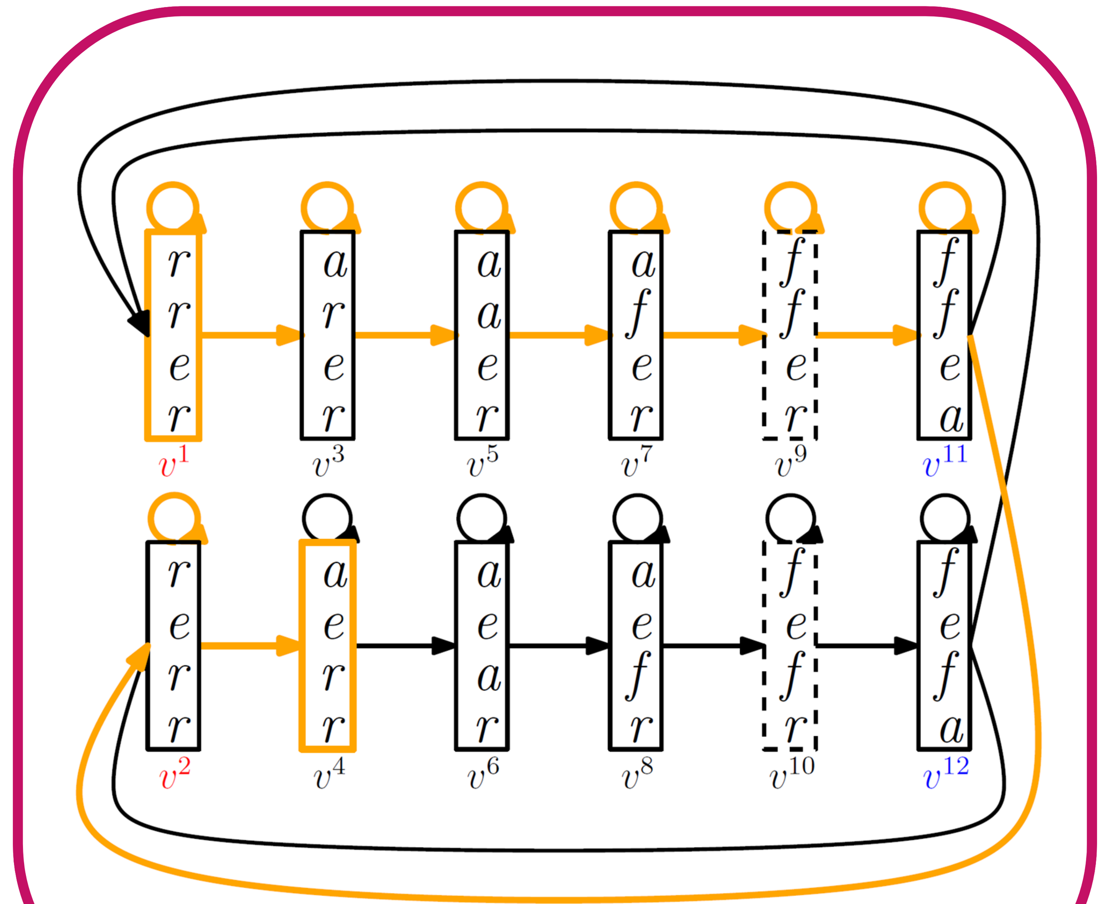
$$\exp\left(-\beta I_b(v^k)\right) \cdot p(O_t|x(v^k)) \cdot \max_{v_{t-1}}\left\{\max_{v_{1:t-2}} f([v_{1:t-2}, v_{t-1}])\right\}$$

  - Keeping only one best interpretation per each flow node is sufficient to track the globally optimal solution.

$$V_t(v^k) = \left[\arg\max_{v_{1:t-1}} f([v_{1:t-1}, v_t = v^k]), v_t = v^k\right]$$

$$\hat{X}_t = x\left(\arg\max_{v_{1:t}} f(v_{1:t})\right) = x\left(\arg\max_{V_t(v^k)} f(V_t(v^k))\right)$$

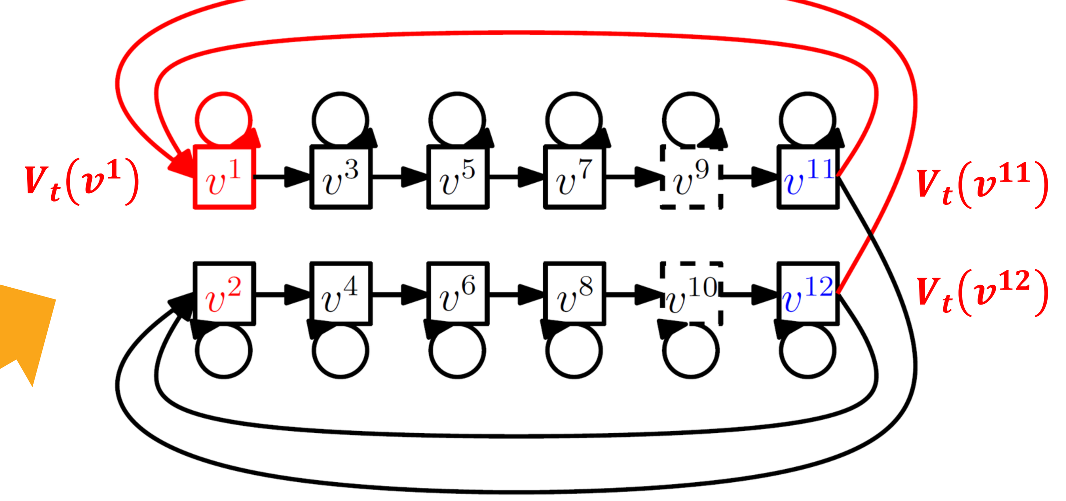
  - The size of the search space is not a parameter anymore, but bounded by the number of flow vertices.



[The constraint flow of "(A^B|C)<D"]

Recognition Example

- $\{V_t(v^k)\}_{k=1}^{12}$ and $\{f(V_t(v^k))\}_{k=1}^{12}$ are given.



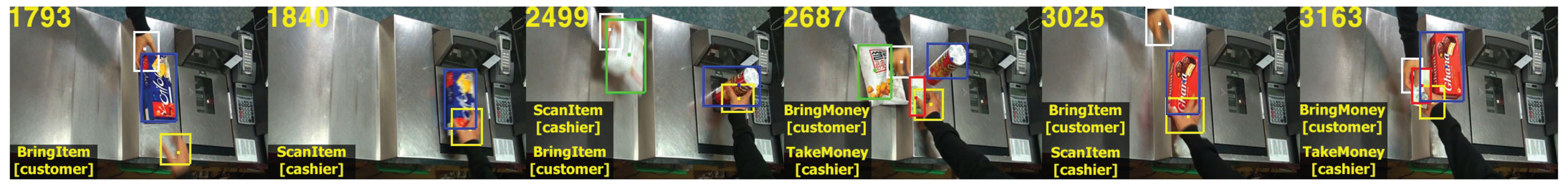
- 3 incoming traces for v^1 at time $t+1$:

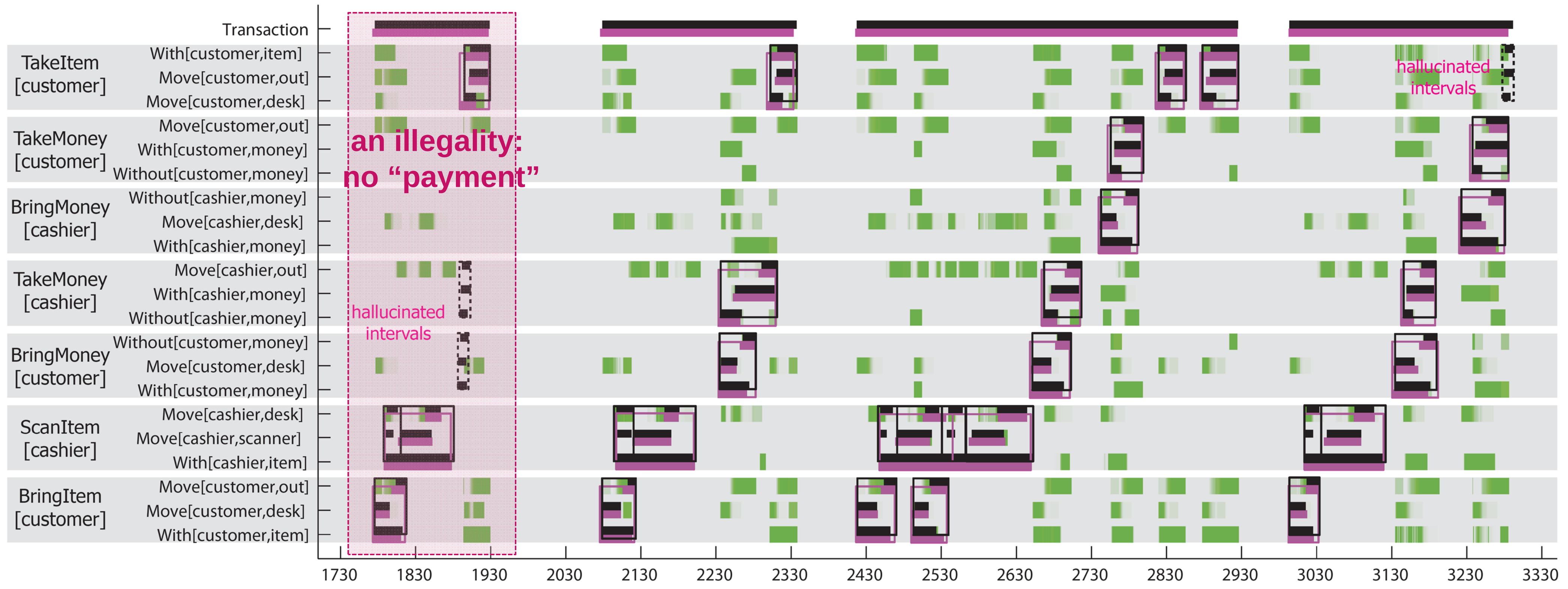
  From v^1: $[V_t(v^1), v^1]$
  From v^{11}: $[V_t(v^{11}), v^1]$
  From v^{12}: $[V_t(v^{12}), v^1]$

  Select the best one among them for $V_{t+1}(v^1)$
- Calculate $f(V_{t+1}(v^1))$ using own observation likelihood and the idle penalty.
- Do the same for the remainders, $\{V_{t+1}(v^k)\}_{k=2}^{12}$

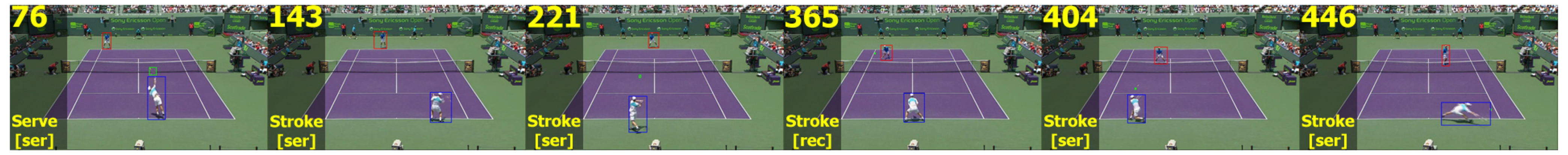
Experiments

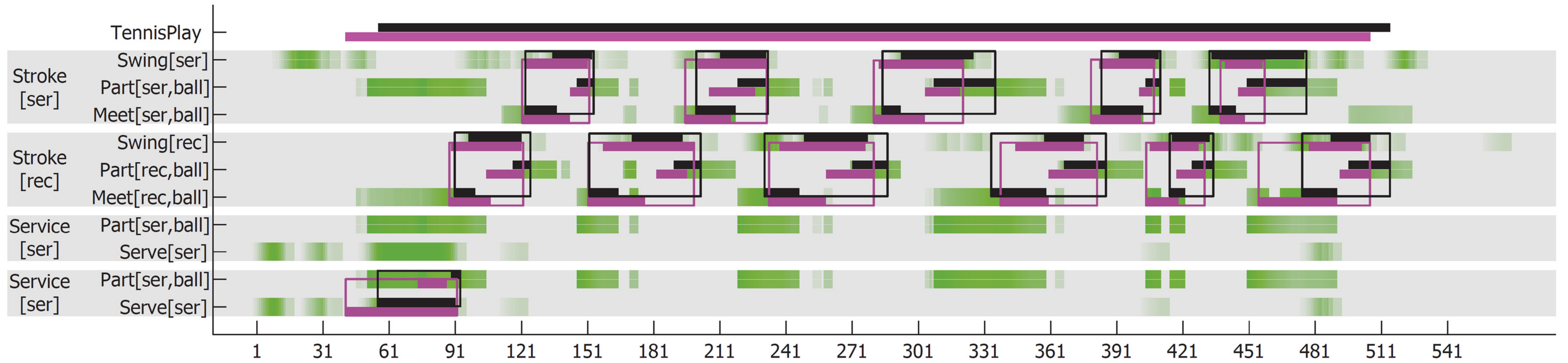
- ***Surveillance sequence***
  - 8 target events (transactions) in a sequence, 476 flow vertices (precision: 0.90, FPR: 0.0053)





- ***Tennis sequence***
  - With action recognizers [Tran *et al*., ECCV'08], 266 flow vertices (precision: 0.69, FPR 0.033)





Conclusion

- A new scenario description method
- Exact inference, which guarantees the optimal solution, via **constraint flow**

Acknowledgement:

This research was supported in part by the IT R&D program of MKE/IITA (2008-F-031-01), and in part by Basic Science Research Program through NRF of Korea funded by MEST (2010-0003496).