5.331 relaxed_sliding_sum

DESCRIPTION LINKS GRAPH

Origin CHIP

Constraint relaxed_sliding_sum(ATLEAST, ATMOST, LOW, UP, SEQ, VARIABLES)

Arguments

ATLEAST	:	int
ATMOST	:	int
LOW	:	int
UP	:	int
SEQ	:	int
VARIABLES	:	collection(var−dvar)

Restrictions

$\texttt{ATLEAST} \geq 0$
$\texttt{ATMOST} \geq \texttt{ATLEAST}$
$\texttt{ATMOST} \leq |\texttt{VARIABLES}| - \texttt{SEQ} + 1$
$\texttt{UP} \geq \texttt{LOW}$
$\texttt{SEQ} > 0$
$\texttt{SEQ} \leq |\texttt{VARIABLES}|$
required(VARIABLES, var)

Purpose There are between ATLEAST and ATMOST sequences of SEQ consecutive variables of the collection VARIABLES such that the sum of the variables of the sequence is in [LOW, UP].

Example $(3, 4, 3, 7, 4, \langle 2, 4, 2, 0, 0, 3, 4\rangle)$

Within the sequence 2 4 2 0 0 3 4 we have exactly 3 subsequences of SEQ = 4 consecutive values such that their sum is located within the interval [LOW, UP] = [3, 7]: subsequences 4 2 0 0, 2 0 0 3 and 0 0 3 4. Consequently the relaxed_sliding_sum constraint holds since the number of such subsequences is located within the interval [ATLEAST, ATMOST] = [3, 4].

Typical

$\texttt{SEQ} > 1$
$\texttt{SEQ} < |\texttt{VARIABLES}|$
$\texttt{range}(\texttt{VARIABLES.var}) > 1$
$\texttt{ATLEAST} > 0 \vee \texttt{ATMOST} < |\texttt{VARIABLES}| - \texttt{SEQ} + 1$

Symmetries

- ATLEAST can be decreased to any value ≥ 0.
- ATMOST can be increased to any value $\leq |\texttt{VARIABLES}| - \texttt{SEQ} + 1$.
- Items of VARIABLES can be reversed.

Algorithm [30].

See also **hard version:** sliding_sum.

used in graph description: sum_ctr *(the sliding constraint)*.

Keywords

characteristic of a constraint: hypergraph.

combinatorial object: sequence.

constraint type: sliding sequence constraint, soft constraint, relaxation.

Arc input(s)	VARIABLES
Arc generator	$PATH \mapsto$collection
Arc arity	SEQ
Arc constraint(s)	• sum_ctr(collection, $\geq$, LOW) • sum_ctr(collection, $\leq$, UP)
Graph property(ies)	• **NARC**$\geq$ ATLEAST • **NARC**$\leq$ ATMOST

Graph model Parts (A) and (B) of Figure 5.661 respectively show the initial and final graph associated with the **Example** slot. For each vertex of the graph we show its corresponding position within the collection of variables. The constraint associated with each arc corresponds to a conjunction of two sum_ctr constraints involving 4 consecutive variables. In Part (B), we did not put vertex 1 since the single arc constraint that mentions vertex 1 does not hold (i.e., the sum $2 + 4 + 2 + 0 = 8$ is not located in interval $[3, 7]$). However, the directed hypergraph contains 3 arcs, so the relaxed_sliding_sum constraint is satisfied since it was requested to have between 3 and 4 arcs.

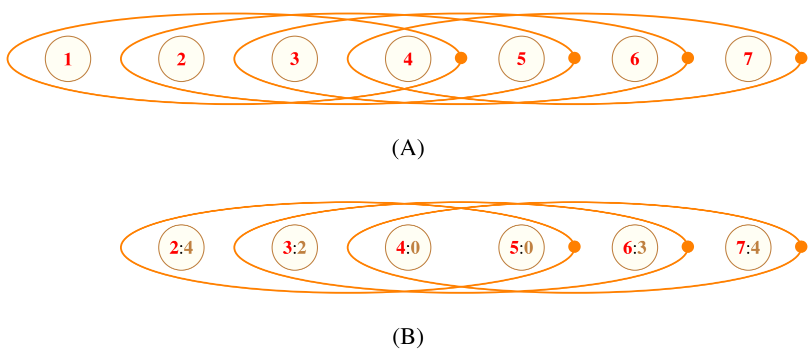

Figure 5.661: (A) Initial and (B) final graph of the relaxed_sliding_sum$(3, 4, 3, 7, 4, \langle 2, 4, 2, 0, 0, 3, 4\rangle)$ constraint of the **Example** slot where each ellipse represents an hyperedge involving SEQ $= 4$ vertices (e.g., the rightmost ellipse represents the constraint $0 + 0 + 3 + 4 \in [3, 7]$)