Revisiting Neighborhood Inverse Consistency on Binary CSPs

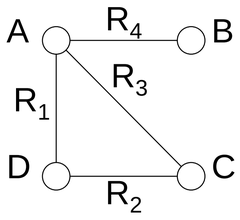
R.J.Woodward[1], S.Karakashian[1], B.Y.Choueiry[1], and C.Bessiere[2]
[1]Constraint Systems Laboratory • University of Nebraska-Lincoln • USA
[2]LIRMM-CNRS • University of Montpellier • France

1. Contributions

1. Understand the structure of the dual graph of a *binary* CSP
2. Determine the impact of this structure on consistency properties, e.g., NIC, sCDC, & RNIC are incomparable
3. Experimentally demonstrate the benefits of higher-level consistency

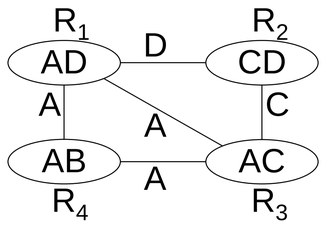
2. Graphical Representation

Constraint Graph



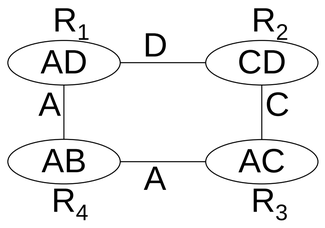
- Variables ⇝ Vertices
- Constraints ⇝ Edges

Dual Graph



- Constraints ⇝ Vertices
- Scope overlap ⇝ Edges

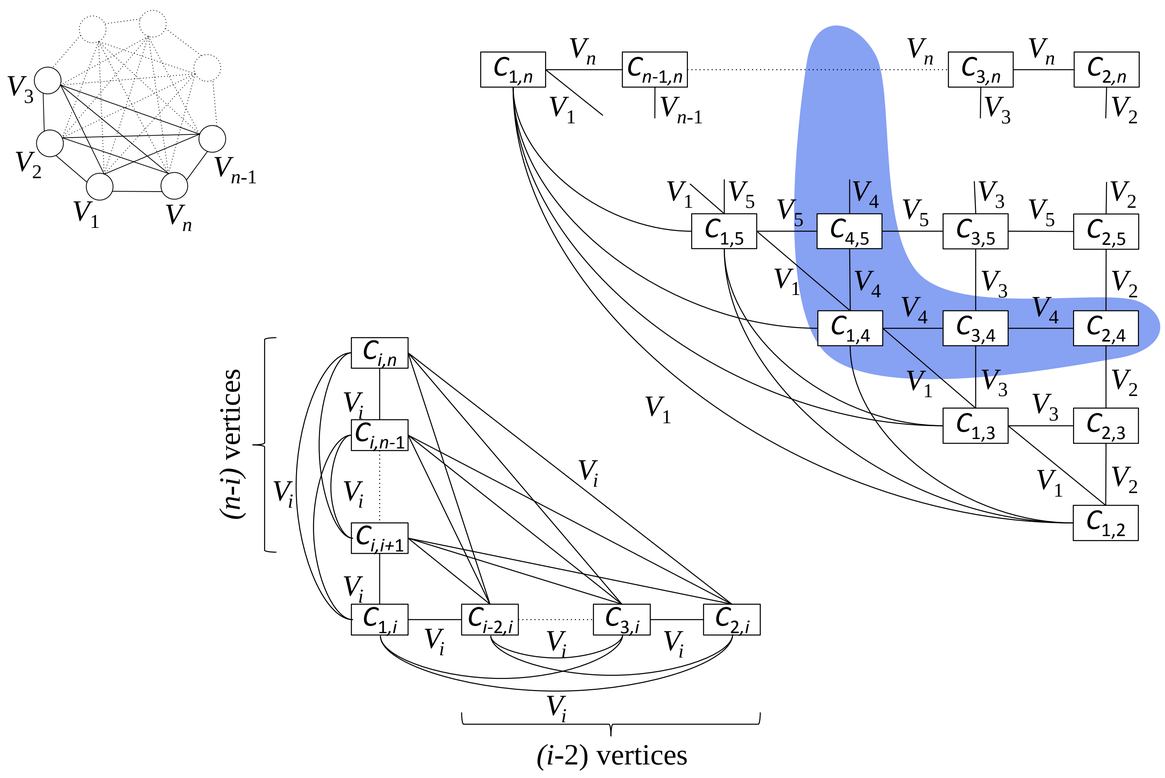
A Minimal Dual Graph



After removing redundant edges [Janssen+, 1989]

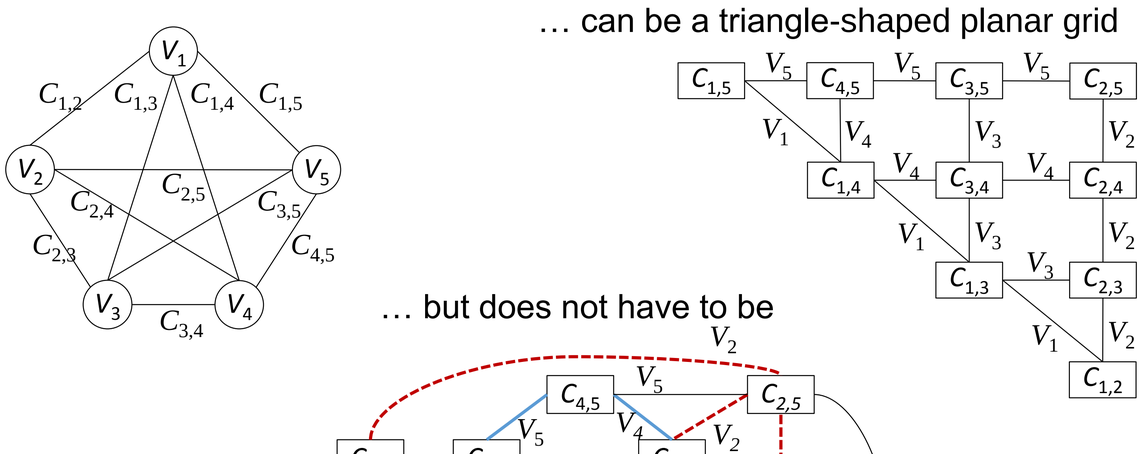
3. Structure of Dual Graph of Binary CSPs

Case1: A Complete Constraint Graph

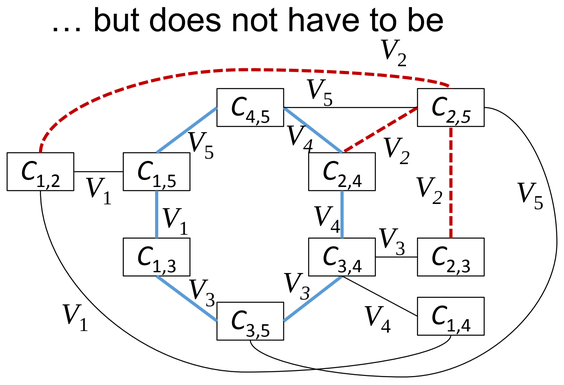


Case1: A Minimal Dual Graph of a Complete Constraint Graph

… can be a triangle-shaped planar grid

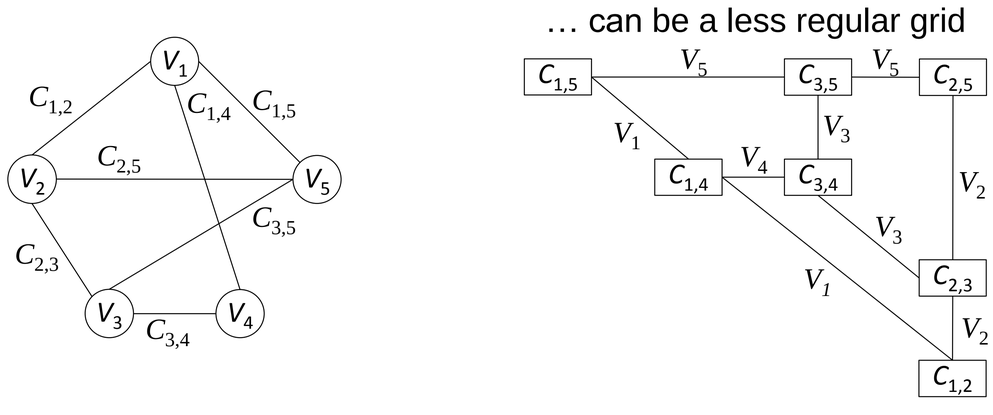


… but does not have to be



Case 2: Non-Complete Constraint Graph

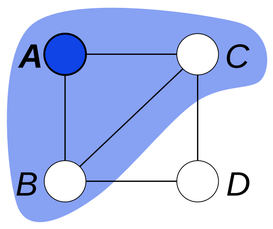
… can be a less regular grid



4. Impact on Local Consistency

NIC, sCDC, and RNIC are not comparable

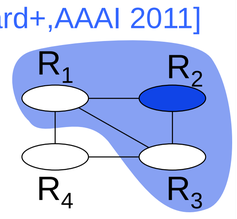
Neighborhood Inverse Consistency (NIC) ensures that every value in the domain of a variable can be extended to a solution in the subproblem induced by the variable and its neighborhood [Freuder & Elfe, AAAI 1996]



Strong Conservative Dual Consistency (sCDC) An instantiation {(*x*,*a*),(*y*,*b*)} is sCDC iff (*y*,*b*) holds in SAC when *x*=*a* and (*x*,*a*) holds in SAC when *y*=*b* and (*x*,*y*) in scope of some constraint, and the problem is AC [Lecoutre+, JAIR 2011]

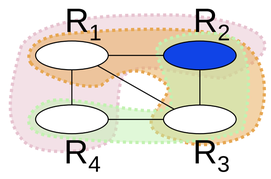
Relational Neighborhood Inverse Consistency (RNIC) ensures that every tuple in every relation R_i can be extended to a solution in the subproblem induced on the dual CSP by $\{R_j\} \cup \text{Neigh}(R_i)$ [Woodward+,AAAI 2011]

- wRNIC, triRNIC, wtriRNIC enforce RNIC on a minimal, triangulated, and minimal triangulated dual graph, respectively
- **selRNIC** automatically selects the RNIC variant based on the density of the dual graph

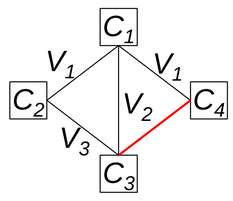
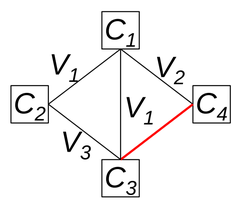


wRNIC is never strictly stronger than R(*,3)C

R(*,*m*) ensures that subproblem induced in the dual CSP by every connected combination of *m* relations is minimal [Karakashian+, AAAI 2010]



wRNIC can never consider more than 3 relations simultaneously

Either  or 

In either case, it is not possible to have an edge between C_3 & C_4 (a common variable to C_3 & C_4) while keeping C_3 as a binary constraint

Experimental Results

Benchmark	# inst.	AC3.1	sCDC1	NIC	selRNIC
		CPU Time (msec)			
		NIC Quickest			
bqwh-16-106	100/100	3,505	3,860	**1,470**	3,608
bqwh-18-141	100/100	68,629	82,772	**38,877**	77,981
coloring-sgb-queen	12/50	680,140	(+3) -	**(+9) 57,545**	634,029
coloring-sgb-games	3/4	41,317	33,307	**(+1) 860**	41,747
rand-2-23	10/10	1,467,246	1,460,089	**987,312**	1,171,444
rand-2-24	3/10	**567,620**	677,253	**(+7)** 3,456,437	677,883
		sCDC1 Quickest			
driver	2/7	(+5) 70,990	**(+5) 17,070**	358,790	(+4) 185,220
ehi-85	87/100	(+13) 27,304	**(+13) 573**	513,459	(+13) 75,847
ehi-90	89/100	(+11) 34,687	**(+11) 605**	713,045	(+11) 90,891
frb35-17	10/10	41,249	**38,927**	179,763	73,119
		RNIC Quickest			
composed-25-1-25	10/10	226	335	1,457	**114**
composed-25-1-2	10/10	223	283	1,450	**88**
composed-25-1-40	9/10	(+1) 288	(+1) 357	120,544	**(+1) 137**
composed-25-1-80	10/10	223	417	(+1) -	**190**
composed-75-1-25	10/10	2,701	1,444	363,785	**305**
composed-75-1-2	10/10	2,349	1,733	48,249	**292**
composed-75-1-40	7/10	(+1) 1,924	(+3) 1,647	631,040	**(+3) 286**
composed-75-1-80	10/10	1,484	1,473	(+1) -	**397**

Benchmark	# inst.	AC3.1	sCDC1	NIC	selRNIC	AC3.1	sCDC1	NIC	selRNIC
		BT-Free				#NV			
		NIC Quickest							
bqwh-16-106	100/100	0	3	**8**	5	1,807	1,881	**739**	1,310
bqwh-18-141	100/100	0	0	**1**	0	25,283	25,998	**12,490**	22,518
coloring-sgb-queen	12/50	1	0	**16**	1	91,853	-	**15,798**	91,853
coloring-sgb-games	3/4	1	1	**4**	1	14,368	14,368	**40**	14,368
rand-2-23	10/10	0	0	**10**	0	471,111	471,111	**12**	471,111
rand-2-24	3/10	0	0	**10**	0	222,085	222,085	**24**	222,085
		sCDC1 Quickest							
driver	2/7	1	**2**	1	1	3,893	**409**	3,763	3,763
ehi-85	87/100	0	**100**	87	**100**	1,425	**0**	**0**	**0**
ehi-90	89/100	0	**100**	89	**100**	1,298	**0**	**0**	**0**
frb35-17	10/10	**0**	**0**	**0**	**0**	24,491	24,491	24,491	**24,346**
		RNIC Quickest							
composed-25-1-25	10/10	0	**10**	**10**	**10**	153	**0**	**0**	**0**
composed-25-1-2	10/10	0	**10**	**10**	**10**	162	**0**	**0**	**0**
composed-25-1-40	9/10	0	**10**	9	**10**	172	**0**	**0**	**0**
composed-25-1-80	10/10	0	**10**	1	**10**	112	**0**	-	**0**
composed-75-1-25	10/10	0	**10**	**10**	**10**	345	**0**	**0**	**0**
composed-75-1-2	10/10	0	**10**	**10**	**10**	346	**0**	**0**	**0**
composed-75-1-40	7/10	0	**10**	7	**10**	335	**0**	**0**	**0**
composed-75-1-80	10/10	0	**10**	1	**10**	199	**0**	-	**0**

Experiments were conducted on the equipment of the Holland Computing Center at the University of Nebraska-Lincoln. Robert Woodward was partially supported by a National Science Foundation (NSF) Graduate Research Fellowship grant number 1041000. This research is supported by NSF Grant No. RI-111795.

October 3rd, 2012