Redundancy Detection in Configuration Knowledge

Alexander Felfernig[†], Florian Reinfrank[†], Gerald Ninaus[†], and Paul Blazek[‡]

[†]Graz University of Technology, Graz, Austria
[‡]cyLEDGE, Vienna, Austria

Contents

- Definitions
- Sequential
- CoreDiag
- Evaluation

Example Knowledge Base

- $Variables\ (V) = \{\texttt{type}, \texttt{fuel}, \texttt{skibag}, 4-\texttt{wheel}, \texttt{pdc}\}$
- $Domains\ (D) = \{dom(\texttt{type}) = \{\texttt{city}, \texttt{limo}, \texttt{combi}, \texttt{xdrive}\},$

  $dom(\texttt{fuel}) = \{4\texttt{l}, 6\texttt{l}, 10\texttt{l}\},$
  $dom(\texttt{skibag}) = \{\texttt{yes}, \texttt{no}\},$
  $dom(4-\texttt{wheel}) = \{\texttt{yes}, \texttt{no}\},$
  $dom(\texttt{pdc}) = \{\texttt{yes}, \texttt{no}\}\}$

- $Knowledge\ Base\ (C_{KB}) = \{$

  $c_1 : 4-\texttt{wheel} = \texttt{yes} \rightarrow \texttt{type} = \texttt{xdrive},$
  $c_2 : \texttt{skibag} = \texttt{yes} \rightarrow \texttt{type} \neq \texttt{city},$
  $c_3 : \texttt{fuel} = 4l \rightarrow \texttt{type} = \texttt{city},$
  $c_4 : \texttt{fuel} = 6\texttt{l} \rightarrow \texttt{type} \neq \texttt{xdrive},$
  $c_5 : \texttt{type} = \texttt{city} \rightarrow \texttt{fuel} \neq 10\texttt{l}\}$

- $Customer\ \ Requirements\ (C_R) = \{$

  $c_6 : 4-\texttt{wheel} = \texttt{no},$
  $c_7 : \texttt{fuel} = 4\texttt{l},$
  $c_8 : \texttt{type} = \texttt{city},$
  $c_9 : \texttt{skibag} = \texttt{no},$
  $c_{10} : \texttt{pdc} = \texttt{yes}\}$

Redundant Knowledge Base

$$C'_{KB} = \{$$

$$c_a : \texttt{skibag} \neq \texttt{no} \rightarrow \texttt{type} = \texttt{limo} \vee \texttt{type} = \texttt{combi} \vee \texttt{type} = \texttt{xdrive},$$

redundant constraint

$$c_1 : \texttt{4-wheel} = \texttt{yes} \rightarrow \texttt{type} = \texttt{xdrive},$$

$$c_2 : \texttt{skibag} = \texttt{yes} \rightarrow \texttt{type} \neq \texttt{city},$$

$$c_3 : \texttt{fuel} = \texttt{4l} \rightarrow \texttt{type} = \texttt{city},$$

$$c_4 : \texttt{fuel} = \texttt{6l} \rightarrow \texttt{type} \neq \texttt{xdrive},$$

$$c_5 : \texttt{type} = \texttt{city} \rightarrow \texttt{fuel} \neq \texttt{10l}\}$$

Redundant Constraint (Definition)

Redundancy can be described as follows: if C = $\{c1, c2, . . . , cn\}$ is a set of constraints and one constraint $ci \in C$ is redundant, then $(C - \{ci\}) \cup$ complement(C) is inconsistent. In this context, complement(C) is the negation of C: if $C = \{c1, c2, . . . , cn\}$ then complement(C) = $\{\neg c1 \vee \neg c2 \vee . . . \vee \neg cn\}$.

Redundant Constraint (Definition)

Definition (Redundant Constraint). Let *ca* be a constraint of the configuration knowledge base *CKB*. *ca* is called redundant iff *CKB* − {*ca*} |= *ca*. If this condition is not fulfilled, *ca* is said to be *nonredundant*. Redundancy can also be analyzed by checking *CKB* − {*ca*} ∪ complement(*CKB*) for consistency. If consistency is given, *ca* is nonredundant.

Minimal Core (Definition)

Definition (Minimal Core). Let *CKB* be a configuration knowledge base. *CKB* is denoted as minimal core iff $\forall ci \in CKB : CKB - \{ci\} \cup$ complement(*CKB*) is consistent. Obviously, $CKB \cup$ complement(*CKB*) $\models \perp$.

Sequential Algorithm for Determining Redundant Constraints

Algorithm 12.1 SEQUENTIAL(C_{KB}): Δ

{C_{KB}: configuration knowledge base}
{$\overline{C_{KB}}$: the complement of C_{KB}}
{Δ: set of redundant constraints}
{C_{KBt}: copy of C_{KB} used for redundancy elimination}
$C_{KBt} \leftarrow C_{KB}$;
for all c_i *in* C_{KBt} **do**
  if $isInconsistent((C_{KBt} - \{c_i\}) \cup \{\neg c_i\})$ **then**
    $C_{KBt} \leftarrow C_{KBt} - \{c_i\}$;
  end if
end for
$\Delta \leftarrow C_{KB} - C_{KBt}$;
$return\ \Delta$;

Execution Trace with SEQUENTIAL

Table 12.1 Example execution trace of SEQUENTIAL. The set of redundant constraints is $\Delta = \{c_a\}$.

SEQUENTIAL Iteration	C_{KBt}	c_i
1	$\{c_a, c_1, c_2, c_3, c_4, c_5\}$	c_a
2	$\{c_1, c_2, c_3, c_4, c_5\}$	c_1
3	$\{c_1, c_2, c_3, c_4, c_5\}$	c_2
4	$\{c_1, c_2, c_3, c_4, c_5\}$	c_3
5	$\{c_1, c_2, c_3, c_4, c_5\}$	c_4
6	$\{c_1, c_2, c_3, c_4, c_5\}$	c_5

CoreDiag

Algorithm 12.2 CoreDiag (C_{KB}): Δ

$\{C_{KB} = \{c_1, c_2, ..., c_n\}\}$
$\{\overline{C_{KB}}$: the complement of $C_{KB}\}$
$\{\Delta$: set of redundant constraints$\}$
$\overline{C_{KB}} \leftarrow \{\neg c_1 \vee \neg c_2 \vee ... \vee \neg c_n\}$;
$return(C_{KB} - \text{CoreD}(\overline{C_{KB}}, \overline{C_{KB}}, C_{KB}))$;

Algorithm 12.3 CoreD$(B, D, C = \{c_1, c_2, ..., c_p\})$: Δ

{B: consideration set}
{D: constraints added to B}
{C: set of constraints to be checked for redundancy}
if $D \neq \emptyset$ *and inconsistent*(B) **then**
  $return\ \emptyset$;
end if
if $singleton(C)$ **then**
  $return(C)$;
end if
$k \leftarrow \lceil \frac{p}{2} \rceil$;
$C_1 \leftarrow \{c_1, c_2, ..., c_k\}$;
$C_2 \leftarrow \{c_{k+1}, c_{k+2}, ..., c_p\}$;
$\Delta_1 \leftarrow \text{CoreD}(B \cup C_2, C_2, C_1)$;
$\Delta_2 \leftarrow \text{CoreD}(B \cup \Delta_1, \Delta_1, C_2)$;
$return(\Delta_1 \cup \Delta_2)$;

Performance Evaluation

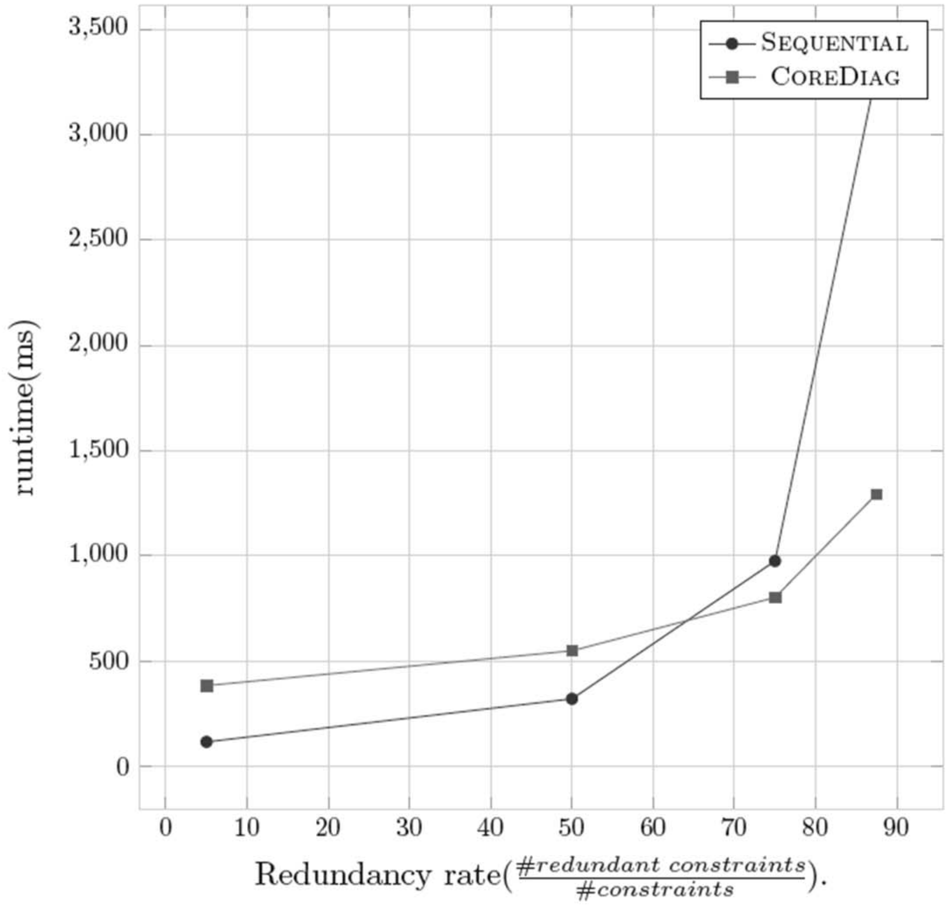

FIGURE 12.1

Performance of SEQUENTIAL and COREDIAG for a financial services knowledge base (see Felfernig et al. 2011).

Exercises

1. Develop a redundancy-free CSP-based configuration knowledge base.
2. Include two redundant constraints.
3. Show the identification of these two redundant constraints on the basis of SEQUENTIAL.

Thank You!

References (1)

(1) Bakker, R., Dikker, F., Tempelman, F., Wogmim, P., 1993. Diagnosing and solving over-determined constraint satisfaction problems. In: 13th International Joint Conference on Artificial Intelligence, Chambery, France, pp. 276–281.

(2) Belov, A., Janota, M., Lynce, I., Marques-Silva, J., 2012. On computing minimal equivalent subformulas. In:Principles and Practices of Constraint Programming (CP 2012). No. 7514 in Lecture Notes in Computer Science. Springer, Québec City, QC, Canada, pp. 158–174.

(3) Boufkhad, Y., Roussel, O., 2000. Redundancy in random SAT formulas. In: 17th National Conference on Artificial Intelligence (AAAI-00), Austin, Texas, pp. 273–278.

(4) Chklovski, T., Gil, Y., 2005. An analysis of knowledge collected from volunteer contributors. In: 20th National Conference on Artificial Intelligence (AAAI-05), Pittsburg, Pennsylvania, pp. 564–571.

(5) Chmeiss, A., Krawczyk, V., Sais, L., 2008. Redundancy in CSPs. In: Ghallab, M., Spyropoulos, C.D., Fakotakis, N., Avouris, N.M. (Eds.), 18th European Conference on Artificial Intelligence (ECAI 2008), Patras, Greece, pp. 907–908.

(6) Dechter, A., Dechter, R., 1987. Removing redundancies in constraint networks. In: 6th National Conference on Artificial Intelligence (AAAI-87), Seattle, Washington, pp. 105–109.

(7) Fahad, M., Qadir, M., 2008. A framework for ontology evaluation. In: Supplementary Proceedings of the 16th Intl. Conference on Conceptual Structures (ICCS 2008), Toulouse, France, pp. 149–158.

(8) Felfernig, A., Burke, R., 2008. Constraint-based recommender systems: Technologies and research issues. ACM Intl. Conference on Electronic Commerce (ICEC08), Innsbruck, Austria, pp. 17–26.

(9) Felfernig, A., Friedrich,G., Jannach,D., Stumptner, M., 2004. Consistency-based diagnosis of configuration knowledge bases. Artificial Intelligence 152 (2), 213–234.

(10) Felfernig, A., Zehentner, C., Blazek, P., 2011. CoreDiag: Eliminating redundancy in constraint sets. In: 22nd International Workshop on Principles of Diagnosis (DX'2011), Murnau, Germany, pp. 219–224.

References (2)

(11) Felfernig, A., Schubert, M., Zehentner, C., 2012. An efficient diagnosis algorithm for inconsistent constraint sets. Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AI EDAM) 26 (1), 53–62.

(12) Felfernig, A., Reiterer, S., Reinfrank, F., Ninaus, G., Jeran, M., 2014. Conflict detection and diagnosis in configuration. In: Felfernig, A., Hotz, L., Bagley, C., Tiihonen, J. (Eds.), Knowledge-based Configuration – From Research to Business Cases. Morgan Kaufmann Publishers, Waltham, MA, pp. 73–87 (Chapter 7).

(13) Fourdrinoy, O., Grégoire, E., Mazure, B., Saïs, L., 2007. Eliminating redundant clauses in SAT instances. In: 4th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. No. 4510 in Lecture Notes in Computer Science. Springer, Brussels, Belgium, pp. 71–83.

(14) Friedrich, G., Shchekotykhin, K., 2005. A general diagnosis method for ontologies. In: 4th International Semantic Web Conference (ISWC05). LNCS 3729, Springer, Galway, Ireland, pp. 232–246.

(15) Friedrich, G., Jannach, D., Stumptner, M., Zanker, M., 2014. Knowledge engineering for configuration systems. In: Felfernig, A., Hotz, L., Bagley, C., Tiihonen, J. (Eds.), Knowledge-based Configuration – From Research to Business Cases. Morgan Kaufmann, Waltham, MA, pp. 139–155 (Chapter 11).

(16) Grimm, S., Wissmann, J., 2011. Elimination of redundancy in ontologies. In: The Semantic Web: Research and Applications, 8th Extended SemanticWeb Conference (ESWC 2011). No. 6643 in Lecture Notes in Computer Science. Springer, Heraklion, Greece, pp. 260–274.

(17) Hotz, L., Felfernig, A., Stumptner, M., Ryabokon, A., Bagley, C., Wolter, K., 2014. Configuration knowledge representation and reasoning. In: Felfernig, A., Hotz, L., Bagley, C., Tiihonen, J. (Eds.), Knowledge-based Configuration – From Research to Business Cases. Morgan Kaufman Publishers, Waltham, MA, pp. 41–72 (Chapter 6).

References (3)

(18) Junker, U., 2004. QUICKXPLAIN: Preferred explanations and relaxations for over-constrained problems. In: McGuinness, D.L., Ferguson, G. (Eds.), 19th International Conference on Artifical Intelligence (AAAI'04). AAAI Press, pp. 167–172.

(19) Levy, A., Sagiv, Y., 1992. Constraints and redundancy in Datalog. In: 11th Symposium on Principles of database systems, San Diego, California, pp. 67–80.

(20) Liberatore, P., 2005. Redundancy in logic I: CNF propositial formulae. Artificial Intelligence 163 (2), 203–232.

(21) Liberatore, P., 2008a. Redundancy in logic II: 2CNF and horn propositional formulae. Artificial Intelligence 172 (2–3), 265–299.

(22) Liberatore, P., 2008b. Redundancy in logic III: Non-monotonic reasoning. Artificial Intelligence 172 (11), 1317–1359.

(23) Piette,C., 2008. Let the solver deal with redundancy. In: 20th IEEE InternationalConference on Toolswith Artificial Intelligence (ICTAI'08), Dayton, Ohio, pp. 67–73.

(24) Richardson, M., Domingos, P., 2003. Building large knowledge bases by mass collaboration. In: 2nd InternationalConference on Knowledge Capture (K-CAP 2003), Sanibel Island, Florida, pp. 129–137.

(25) Sabin, M., Freuder, E. 1999. Detecting and resolving inconsistency and redundancy in conditional constraint satisfaction problems. AAAI 1999Workshop on Configuration (AAAI Technical ReportWS-99-05), Orlando, Florida, 90–94.