A Constraint Satisfaction Approach to Data-Driven Implementation of Clinical Practice Guidelines

Craig Kuziemsky, PhD[1], Dympna O'Sullivan, PhD[1],Wojtek Michalowski, PhD[1], Szymon Wilk, PhD[1], Ken Farion, MD[2]

[1] Telfer School of Management, University of Ottawa, Ottawa, Canada
[2] Faculty of Medicine, University of Ottawa, Ottawa, Canada

Abstract

Despite significant research efforts, the implementation of computerized clinical practice guidelines (CPG) in practice remains problematic for a number of reasons. In particular most guideline representation models do not deal adequately with incomplete or inconsistent clinical data. We present a constraint satisfaction approach to address such shortcomings by focusing on CPG data rather than CPG representation. We model a CPG as a set of data-driven constraints which are used to generate complete solutions for describing a patient state from incomplete clinical data, where the patient state is confirmed by the user. Inconsistent input data can be temporarily eliminated and final feasible solutions (permitted complete solutions from a CPG) can pinpoint inconsistencies in original input data alongside allowable guideline data. We demonstrate a sample implementation of the approach for a pediatric asthma CPG.

Introduction

Research into the implementation of CPG in clinical practice has focused on two main topics. First, methodologies have been developed for facilitating the translation of natural language guideline documents (medical literature) to computer interpretable formats[1]. Second, a large body of research has focused on developing formal, structured, and common representation models for guideline data to facilitate interpretation in multiple implementation environments[2]. Examples of such models include Asbru, PROforma and the Guideline Interchange Format (GLIF). While this research has progressed computer based modeling, authoring, and dissemination of guidelines, solutions that actually implement guidelines in clinical practice remain exceptional. A key limitation in CPG models is they overlook issues on how guidelines are actually used in practice. For instance, focusing on representation tends to emphasize population-oriented or general recommendations whereas implementing guidelines in practice requires more patient-centered or data-driven suggestions[3]. Furthermore representation models encounter implementation problems with incomplete or problematic input data. However, in practice, clinical data is routinely collected in an inconsistent, incomplete and/or inexact manner[4].

Current guideline representation models are represented as primitives including action steps, (clinical task to be performed or avoided), decision steps (selection from a set of alternatives) and patient state (describing clinical status of a patient) [2]. Primitives can be represented using rules (e.g., Arden Syntax rules in GLIF), as workflow systems, or models such as Bayesian models (see www.openclinical.org for examples of such approaches). A common challenge with primitive representation is they are sensitive to data that is missing or mismatched with the CPG, causing problems when matching rules against patient data. This issue can be addressed using rule inferences that allow for partial matching. However, the outcome of such matching schema is usually some 'estimate of strength' or score which may not be useful as no insight is provided into what data is used in performing calculations. In practice it would be more useful to leverage possible data-based outcomes to allow a physician to check to what extent such an outcome corresponds to data at hand (i.e. patient data already collected).

In this paper we address these issues using a flexible data-driven approach to CPG implementation, where a CPG is cast as a constraint satisfaction model with constraints defining allowable variable values and permitted combinations of values for clinical variables from a CPG. Incomplete clinical data is input to a constraint model and extended to complete CPG solutions for describing a patient state. Modeling a CPG in such a manner can overcome limitations of current guideline models such as missing or mismatched data. At present our constraint satisfaction approach is only concerned with expanding the decision step of a guideline model in order to provide a more comprehensive representation of the decision to be undertaken. Once the decision step has been implemented using the constraint-based approach it could be easily extended to automatically trigger the action step of a CPG which could provide the user with a list of possible recommendations for the decision and patient state. The paper is organized as follows. First we provide an overview of constraint satisfaction. Second we describe our framework for implementing CPG as a constraint model. Third we present an implementation of the approach for a pediatric asthma CPG. Finally we conclude with a discussion.

Constraint Satisfaction

Constraint satisfaction solves problems by stating constraints about the problem area and consequently finding solutions that satisfy all or some of these constraints. A constraint satisfaction problem is defined by a tuple $P = (X, D, C)$ where $X = \{X_1,\ldots,X_n\}$ is a finite set of variables, each associated with a domain of discrete values $D = \{D_1,\ldots,D_n\}$, and a set of constraints $C = \{C_1,\ldots C_n\}$. Each constraint C_i is expressed by a relation R_i on some subset of variable values, $R_i \subseteq D_{i1} \times \ldots \times D_{ik}$ and denotes the tuples that satisfy C_i [5]. A solution to a constraint satisfaction problem is an assignment of domain values to variables, in such a way that constraints are satisfied and solutions are found by systematically searching through the space of possible assignments of values to variables. This may involve finding just one solution with no preferences, all solutions or an optimal solution given some objective function determined in terms of some or all variables.

In this work we make use of two commonly used constraint satisfaction methodologies for finding feasible solutions to CSP constraint models. Firstly we employ a *Backtracking* search methodology to extend an incomplete input solution to a complete solution. The backtracking search methodology attempts to incrementally extend an incomplete solution towards a complete solution, by repeatedly choosing values for variables which are consistent with values in the incomplete solution. If an incomplete solution violates any constraints, backtracking is performed to the most recently instantiated variable that still has alternatives available. The search guarantees a solution, if one exists, by searching for all possible solutions from the input data, or else proves that the problem is unsolvable. We also employ a variation of a constraint satisfaction methodology known as *Node Consistency. Node Consistency* is a technique for detecting inconsistent values in an input that cannot lead to a solution. The technique we use works as follows: A node representing a variable X is node consistent if for a value d in the domain of X, a unary constraint (a constraint defined on a single domain variable) on X is satisfied. If the domain D of a variable X contains a value "d" that does not satisfy a unary constraint on X, then the instantiation of X to "d" will result in failure. Thus, the node inconsistency can be eliminated by simply removing those values from the domain D of each variable X that do not satisfy the unary constraint on X.

A CPG Constraint Satisfaction Approach

For a given disease, CPG define a number of possible diagnoses in terms of clinical attributes and their expected values. Each CPG diagnosis can be viewed as a decision step and we refer to this as a diagnosis decision step (DDS) hereafter. The constraint satisfaction approach presented in this paper represents a CPG DDS as a constraint model by defining appropriate attributes and values and relations between them for the DDS; identifying possible feasible solutions for the constraint model; and defining a methodology for generating solutions to the constraint model. Given a constraint model representing a DDS, a confirmed diagnosis from a physician and an incomplete set of clinical values describing a patient state, the aim of the constraint satisfaction approach is to identify a complete set of feasible values from the CPG for describing the patient state. A feasible solution to the problem is any DDS solution or outcome involving an allowable combination of all possible attribute values for the particular diagnosis that also includes all consistent values from the incomplete input solution. The solution methodology is a backtracking search that finds all possible complete solutions, where feasible solutions include all node consistent values from the incomplete input solution.

Therefore, using the constraint satisfaction approach one outcome of a DDS from a CPG may be modeled as a tuple $CPG\_{DDSi} = (Xi, Di, Ci)$, where $Xi = \{Xi_1,\ldots,Xi_n\}$ is a finite set of clinical attributes, $Di = \{Di_1,\ldots,Di_n\}$, are the domains of the attributes, i.e. a set of attribute values and $Ci = \{Ci_1,\ldots,Ci_n\}$ is a set of constraints defining either allowable individual attribute value pairs (unary constraints) for a given DDS, or permitted relations among all attribute values (non-unary constraints) for a DDS. (It is possible to have many outcomes/solutions for one DDS as well as many decision steps in a CPG.)

The first types of constraints are referred to as *partial-patient-descriptor-constraints* and define all possible values for individual clinical attributes in a DDS. More formally a unary constraint for a DDS_i is:

$$C_i = \{X_i \{Z_i\}\}, \text{ where } Z_i \subseteq Di \text{ and } Z_i \neq \phi$$

These unary constraints are used to sanity check individual attribute values from an incomplete input solution to ensure node consistency. The second set of constraints is called *complete-patient-descriptor-constraints*. These constraints model allowable relations over all possible attribute values for a given DDS (combined using a Boolean AND) and are used to find feasible complete solutions for describing a patient state by extending a node consistent incomplete description of a patient state. For a DDS_i, these constraints are defined by:

$$C_i = \{X_{i1} \{Z_i\} \wedge X_{i2} \{Z_i\} \wedge \ldots. \wedge X_{in}(Z_i)\},$$
$$\text{where } Z_i \subseteq D_i \text{ and } Z_i \neq \phi$$

A feasible solution to a DDS constraint model is any complete set of values for describing patient state, which fully satisfy any constraint from the set of *complete-patient-descriptor-constraints* and which includes all values from an incomplete input solution that satisfy any constraints from the set of *partial-patient-descriptor-constraints*. Our constraint satisfaction approach for solving a problem is as follows:

Step 1: Perform node consistency check on incomplete input solution - A confirmed diagnosis and an incomplete solution (set of partial clinical data) provided by a physician as he/she assesses a patient are the input to the constraint satisfaction model. The DDS constraint model matching the diagnosis supplied by the physician is invoked and a node-consistency check is performed on each value from the incomplete input using the set of *partial-patient-descriptor-constraints*. There are three possible outcomes to the node consistency check – no consistent nodes are found, all nodes are found to be consistent, or the result may be a combination of consistent and inconsistent nodes. If no consistent nodes are found (i.e. all *partial-patient-descriptor-constraints* are violated), the physician is informed that all data entered is inconsistent for the confirmed diagnosis. If all nodes are consistent (i.e. all *partial-patient-descriptor-constraints* are satisfied), the incomplete solution may be extended to a feasible solution for the particular DDS. Therefore the input solution is retained as is, and provided as an input to the *complete-patient-descriptor-constraints* (step 2). Finally if a combination of consistent and inconsistent nodes are found (i.e. some *partial-patient-descriptor-constraints* are violated and some are satisfied), inconsistent values are eliminated from the incomplete solution and consistent values are retained. Inconsistent values are referred to as *conflict values*, and are flagged to draw attention to the mismatched CPG and patient data in final solutions. Retained values that have satisfied *partial-patient-descriptor-constraints* are provided as input to the *complete-patient-descriptor-constraints* (step 2).

Step 2: Extend incomplete solution to complete solution - Given a node consistent input solution obtained using step 1, the approach extends the incomplete solution to complete solutions for describing the patient state. Complete solutions are those which contain feasible values for all possible clinical attributes for a DDS from the CPG. Solutions are found using a backtracking search and *complete-patient-descriptor-constraints.* Given the incomplete input, values for missing variables are found that are consistent with values present in the current input solution; variables are instantiated sequentially and when all variables are instantiated, the validity of the constraint is checked using *complete-patient-descriptor-constraints*. The search continues until all possible solutions are found.

Step 3: Order complete solutions - A DDS constraint model may return multiple outcomes or feasible solutions for an input solution. Therefore, solutions are ranked using a normalized Euclidean distance measure to calculate the similarity

	Mild	Moderate	Severe
Dyspnea	exertional	at rest	labored
Beta Agonist	good response	partial response	weak or no response
Difficult Speech	absent	absent or moderate	moderate or present
Tachycardia	absent	absent	present
Tachypnea	exertional	exertional or at rest	at rest or labored
Accessory muscles	none	none or moderate	moderate or severe
Breathing sounds	normal or reduced	reduced	reduced or silent
Typical episode	better or same	same or worse	same or worse
Sa02	>95	92-95	<92
PEV	>75	50-75	<50
FEV	>75	50-75	<50

Table 1: CAEP pediatric asthma CPGDSS model

between an infeasible input solution (containing *conflict values)* and feasible solutions returned by the DSS model (where conflict values were replaced by missing values). The measure requires non-numeric attribute values DDS model (where *conflict values* were replaced by missing values). The measure requires non-numeric attribute values to be encoded in a numeric format. However, if no *conflict values* are found in the original input solution, each solution is assigned an equal rank as in the absence of *conflict values* all solutions are equally adequate for representing the patient state. Solutions are presented so that feasible guideline values for *conflict values* as well as actual *conflict values* from input data are displayed to draw attention to mismatched CPG and patient data. The methodology allows the physician to visualize in a data-driven manner, the different effects of mismatched data on the patient state as well as to examine the extent to which they are applying the CPG for the particular patient. The ultimate decision to save or overwrite data describing the final patient state is left to the physician.

Sample Implementation: Pediatric Asthma

We demonstrate our constraint satisfaction approach we apply it to the CAEP (Canadian Association of Emergency Physicians) pediatric asthma guideline as shown in Table 1. The guideline allows physicians to distinguish if a patient is suffering from a mild, moderate or severe asthma exacerbation by analyzing values for 8 different clinical signs (dyspnea - typical episode) and 3 clinical measurements (SaO2, PEV, and FEV). The guideline also consists of recommended treatments that should be applied given different levels of exacerbation severity, however our CPG constraint modeling approach currently focuses only on the decision step component of guideline modeling and thus in Table 1 we omit the guideline component that deals with treatment. To transform the asthma CPG from its representation in Table 1 to one consistent with our constraint satisfaction approach, we define a DDS constraint model for each diagnosis/severity category; mild, moderate, and severe. For each model the set of variables *X,* are the 11 clinical signs and measurements, X = {dyspnea, beta agonist,…,FEV} as shown in the first column of Table 1, and D, the set of domain values are the associated values for those signs and measurements in the columns labeled mild, moderate and severe in Table 1. For example, for the moderate DDS the variable X_3 = {difficult speech} has domain values D_3 = {absent, moderate}.

For each DDS model we define two sets of constraints C; unary *partial-patient-descriptor-constraints* and *complete-patient-descriptor-constraints*. For example unary constraints on the variable ‘difficult speech’ are C_{DDS3} = difficult speech{absent} for the mild DDS constraint model, and $C_{DDS3\{1\}}$=difficult speech{absent} and $C_{DDS3\{2\}}$=difficult speech {moderate} for the moderate DDS constraint model. A sample *complete-patient-descriptor-constraint* for the mild DDS constraint model is: C_{DDS1} = {dyspnea (exertional) Λ beta agonist (good response) Λ difficult speech (absent) Λ tachycardia (absent) Λ tachypnea (exertional) Λ accessory muscles (none) Λ breathing sounds (normal) Λ typical episode (same) Λ SaO2 (>95) Λ PEV (>75) Λ PEV (>75)}.

We demonstrate an application of the CPG constraint satisfaction approach using the sample description of a mild asthma patient shown in Table 2. The input solution is incomplete in that it contains missing data and data that is mismatched with the CPG (e.g. the value ‘worse’ for the variable ‘typical episode’ is mismatched for a diagnosis of mild from the CPG). We demonstrate how the CPG constraint satisfaction approach may extend the incomplete solution in Table 2 to a complete feasible solution for describing the patient state.

Attribute	Value
Dyspnea	exertional
Beta Agonist	good response
Difficult Speech	absent
Tachycardia	absent
Typical Episode	worse

Table 2: Input solution

Step 1: Perform node consistency check on incomplete input solution - The physician has confirmed that the patient is suffering from mild asthma and thus the incomplete solution in Table 2 is passed as an input to the mild DDS constraint model. A node consistency check is performed on all values from the incomplete solution by checking the validity of each unary constraint defined by *partial-patient-descriptor-constraints*. For the sample incomplete solution, *partial-patient-descriptor-constraints* are both satisfied and violated for the mild DDS constraint model. *Conflict values* detected during the node consistency check are flagged and removed, e.g. a value of 'worse' for variable 'typical episode' and all values which satisfy the *partial-patient-descriptor-constraints* are retained in the input solution and the approach proceeds to step 2.

Step 2: Extend incomplete solution to complete solution - The solution methodology is to perform a backtracking search to find any constraints from the *complete-patient-descriptor-constraints* for the mild DDS constraint model which are fully satisfied. The algorithm searches for all feasible complete solutions by instantiating missing variables from the incomplete input solution with values consistent with those already present in the incomplete solution. For the scenario in Table 2, 4 feasible solutions are found.

Step 3: Order complete solutions - Complete solutions from the mild DDS constraint model are ranked by similarity to the incomplete input solution. For example; a *conflict value* of worse for the variable 'typical episode' was detected in the incomplete solution. In calculating the Euclidean distance between the incomplete and complete solutions from step 2, solutions with a value of 'same' for 'typical episode' are closer in distance than solutions with a value of 'better' for 'typical episode' and ranked higher in the returned ordering. In displaying results, consistent values from retrieved solutions for which *conflict values* were detected in the original input are flagged for attention along with the actual *conflict values*. For example, feasible values for the variable 'typical episode' of 'same' and 'better' from complete solutions are flagged and displayed alongside the actual *conflict value* of 'worse'.

Discussion

In this paper we have introduced a data-driven CPG constraint satisfaction approach for implementing CPG in clinical practice. The approach expands the decision step of guideline models by proposing a methodology for implementing CPG in cases of incomplete data or a mismatch between patient data and CPG. Using constraint satisfaction, incomplete or mismatched data may be extended to find feasible complete solutions for describing patient state. We also provide a methodology for data-driven explanation of the differing effects of mismatched clinical data on the final description of a patient state using constraint satisfaction to leverage different patient 'scenarios' which can include/exclude mismatched data as well as allowing physicians to view the extent to which they are applying a guideline for a particular patient. The framework also enhances data quality by drawing attention to mismatched data and by ensuring the collection of all data for patient encounters. Clinicians consulting on this research believe the ability to facilitate decisions based on incomplete or partially incongruent data would be extremely useful, particularly for novice physicians. We will seek further clinical advice as we extend and implement our framework. We intend to extend the framework by further developing the heuristic for ordering feasible solutions. This could be performed by incorporating domain knowledge into the problem solving process. For instance, there may be subsets of attributes/values from a CPG that may be more discriminatory in describing patient state. This domain knowledge could be elicited directly from experts or it could be learned automatically from a case-base of previous incomplete and complete feasible solutions for describing patient states.

Acknowledgements

The support of the Natural Sciences and Engineering Research Council of Canada and the Canadian Institutes of Health Research is gratefully acknowledged.

References

1. Shiffman RN, Michel G, Essaihi A et al; Bridging the guideline implementation gap: A systematic, document-centered approach to guideline implementation. JAMIA, 2004, 11(5): 418-426
2. Wang D, Peleg M, Tu SW, *et al.* Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines: a literature review of guideline representation models. Int. J. Med. Inf. 2002:68:59-70
3. Sonnenberg FA, Hagerty, CG. Computer-interpretable clinical practice guidelines – Where we are and where we are going? AMIA Yearbook 2006:11:45-158
4. Cios KJ, Moore GW. Uniqueness of medical data mining. A.I. in Medicine. 2002:26:1–24
5. Dechter R. Constraint processing. Morgan Kaufmann 2003