Development of an Intelligent System for Fixture Design Using Case-Based Reasoning (CBR) Technique

Vývoj inteligentného systému pre navrhovanie prípravku pomocou techniky CBR (na prípadoch založené uvažovanie)

Djordje Vukelić, Branko Tadić, Janko Hodolič, Peter Križan, Nenad Simeunović

Abstract

The development of a system for machining fixtures design is presented in this paper. The system provides new fixture layout design on the basis of previously designed solutions. The case-based reasoning technique was used for this system development. Additionally, the system provides optimization of fixture layout. Productivity, accuracy and production costs were used as criteria for optimization. The paper presents basic steps of applied methodology, description of particular system segments and system implementation in production industry.

Keywords: fixture, case-based reasoning

Abstrakt

Článok pojednáva o vývoji systému pre navrhovanie obrábacích prípravkov. Systém poskytuje možnosť navrhovania nových konštrukcií prípravkov na základe predchádzajúcich navrhnutých riešení. Pre vývoj tohto systému bola použitá technika CBR (na prípadoch založené uvažovanie). Popísaný systém umožňuje aj optimalizovať konštrukcie prípravkov. Ako optimalizačné kritériá boli použité produktivita (výrobnosť), presnosť a náklady na výrobu prípravku. V článku sú predstavené základné kroky použitej metodiky, popis jednotlivých segmentov systému a príklad jeho implementácie v priemyselnej výrobe.

Kľúčové slová: prípravok, na prípadoch založené uvažovanie

1. Introduction

Automated manufacturing systems require automated fixture design, since - depending on the type of production - fixture design and manufacture can consume up to 20% of the total time for production preparation [7]. Cutting down preparation time allows reduction of total production costs. Computer aided fixture design has evolved in an attempt to counterbalance the negative effects of conventional design. This design paradigm uses computers to partially automate task sequences in fixture design. The basic goal is to generate adequate fixture solution within acceptable time span, while at the same time reducing the subjective influence of the designer and his/hers effort to a minimum [5]. Various techniques and methods have been used to solve the problem of automated fixture design - one of them being the case-based reasoning.

Case-based reasoning is a concept aimed towards solving new fixture design problems based on existing fixture design solutions [1]. The process follows several steps: first there is a novel workpiece for which a fixture has to be designed. The designer searches the so called knowledge base in order to come up with solutions which are most similar to the problem in hand. Next, if found, the existing solution is modified to meet new requirements and a novel fixture design solution is generated [2].

In case-based reasoning the first step is to structure (order) the problem of fixture design. This is accomplished by means of the so called cases. A case is one of the possible ways to represent knowledge. Accordingly, it consists of a set of criteria which encode the information on fixture design requirements, and a set of criteria which encode the information on fixture design solution. The subset of the design requirements set, which is used to search the knowledge base, consists of indices. A single case consists of indexed and non-indexed criteria [3]. The indexing can be performed manually or automatically. Once the case is structured, the data are being normalized, i.e. transformed to allow efficient manipulation and control. Normalization allows the cases to be organized in a way which allows their comparison. It is a process in which a certain set of data is allotted a common attribute. Upon completion of this phase, the data are organized in a controllable way, thus allowing another phase of data aggregation to commence. Here, a principle of similarity (distance) is used, which calculates the distance between the novel (desired) and the old (existing) fixture design solution. All search methods transform several indices into a single value (column), which serves as a criterion for sorting and selection of the closest case. After the aggregation is completed - that is, the system presents the designer with the solutions which can be used for decision-making - the next step is to analyze the design solution. Upon identifying one or several eligible cases, one should check the possibility to employ an existing design solution in order to generate the new fixture. This requires development of mechanisms for modification of existing fixture design solutions to match the problem in hand. The process of modification requires important similarities/ differences to be noted between the novel and the existing solution in order to implement the necessary changes [4].

Case-based reasoning has so far seen successful application in the phase of fixture layout planning, i.e. defining of the locating and clamping surfaces for a predefined layout scheme [8]. However, the phase of detailed fixture design still requires the designer to work interactively with software application in order to select the necessary fixture elements.

The deficiency of such systems is that they provide only concept solutions [1, 3, 6]. The work required to arrive from the concept to the final design solution is left to the designer alone. This can be a time-consuming task, provided that the problem can be solved in the first place. In addition, the problem with the research up-to-date was a lack of clear and systematic definition of criteria which would guide the necessary modifications [8]. None of the previous approaches have either been able to provide selection of applicable finished (existing) fixture solutions, nor to allow design of novel fixture solutions in cases when the system fails to come up with an acceptably similar solution. Regar-