Applying Covering-Based Rough Set Theory to User-Based Collaborative Filtering to Enhance the Quality of Recommendations

Zhipeng Zhang[1], Yasuo Kudo[2(✉)], and Tetsuya Murai[3]

[1] Course of Advanced Information and Electronic Engineering, Graduate School of Engineering, Muroran Institute of Technology, Mizumoto 27-1, Muroran 050-8585, Japan
14096507@mmm.muroran-it.ac.jp

[2] College of Information and Systems, Muroran Institute of Technology, Mizumoto 27-1, Muroran 050-8585, Japan
kudo@csse.muroran-it.ac.jp

[3] Graduate School of Information Science and Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060-0814, Japan
murahiko@ist.hokudai.ac.jp

Abstract. Recommender systems provide personalized information by learning user preferences. Collaborative filtering (CF) is a common technique widely used in recommendation systems. User-based CF utilizes neighbors of an active user to make recommendations; however, such techniques cannot simultaneously achieve good values for accuracy and coverage. In this study, we present a new model using covering-based rough set theory to improve CF. In this model, relevant items of every neighbor are regarded as comprising a common covering. All common coverings comprise a covering for an active user in a domain, and covering reduction is used to remove redundant common coverings. Our experimental results suggest that this new model could simultaneously present improvements in accuracy and coverage. Furthermore, comparing our model with the unreducted model using all neighbors, our model utilizes fewer neighbors to generate almost the same results.

Keywords: Covering-based rough set · Recommender systems · Collaborative filtering · Covering reduction

1 Introduction

With the development of the internet and artificial intelligence (AI), the recommender system (RS) has become very popular recently. The RS can advise users when making decisions on the basis of personal preferences and help users discover items they might not find by themselves [1,2]. Collaborative filtering (CF) is a significant component of the recommendation process [4], that is based on the ways in which humans have made decisions throughout history [7,8]. User-based CF uses information of the active user's neighbors to make predictions

V.-N. Huynh et al. (Eds.): IUKM 2015, LNAI 9376, pp. 279–289, 2015.
DOI: 10.1007/978-3-319-25135-6_27

and recommendations [3]. Accuracy and coverage are two crucial metrics for evaluating the RS, but it is difficult to simultaneously achieve good values for these metrics when using user-based CF to make recommendations.

Rough set theory was first presented by Pawlak in the early 1980s [5]. Covering-based rough set has been regarded as a meaningful extension of the classical rough set to handle vague and imperfect knowledge better, which extends the partition of rough set to a covering [13,14]. The notion of reduction for covering is one of the most important results in covering-based rough set [17]. Currently, much of the literature has been focused on providing the theory behind covering-based rough set [13–16], but there is little regrading applications, especially for RSs.

In this study, covering-based rough set theory is first applied to the RS. We present a new model called covering-based collaborative filtering (CBCF) that uses the covering-based rough set to improve the user-based CF approach. The covering reduction notion is included to extract the relative effective neighbors from all neighbors, thus defining the active user's reduct-neighbors. According to these reduct-neighbors, we obtain predictions and recommendations more efficiently.

The remainder of our paper is organized as follows. In Section 2, we review the basic notions and knowledge of covering-based rough sets and covering reductions. In Section 3, we propose the covering-based rough set model for user-based CF. Next, in Section 4, we present our experiments and compare our results with a model that does not employ the covering reduction as well as a model using CF with the same number of neighbors. Finally, in Section 5, we note our conclusions and define areas for future work.

2 Background

2.1 Covering-Based Rough Set

In this subsection, we present the basic knowledge of covering and the covering approximation space. More details can be found in [13,14].

Definition 1. *Let U be the domain of discourse and C be a family of subsets of U. If none of the subsets in C is empty, and $\cup \mathrm{C} = \mathrm{U}$, C is called a covering of U.*

Definition 2. *Let U be a non-empty set and C be a covering of U. We call the ordered pair $\langle \mathrm{U}, \mathrm{C} \rangle$ a covering approximation space.*

We note here the definition of covering is an extension of the definition of partitions. Different lower and upper approximation operations would generate different types of covering-based rough set. The covering-based rough set was first presented by Zakowski [12], who extended Pawlak's rough set theory from a partition to a covering. Pomykala gave the notion of the second type of covering-based rough set [6], while Tsang presented the third type [9], Zhu defined the fourth [16] and fifth [15] types of covering-based rough set models, and Wang studied the sixth type of covering-based approximations [10].

2.2 Reduction Theory of Covering-Based Rough Set

The reduction of covering concept was presented by Zhu in [17]; Zhu also presented other covering reduction approaches in [15,18]. In [11], Yang initially constructed a new reduction theory by redefining the approximation spaces and the reductions of covering generalized rough set, which was applicable to all types of covering generalized rough set. In our present study, we focus only on one type of covering reduction algorithm, i.e., the one presented by Zhu in [18], because this algorithm could remove redundant neighbors more efficiently. Definition 3 defines this type of covering reduction algorithm, which corresponds to the definition of $exclusion(C)$ in [18].

Definition 3. *Let C be a covering of a domain U and* $\mathrm{K} \in \mathrm{C}$. *If there exists another element* K' *of C such that* $\mathrm{K} \subset K'$, *we say that K is reducible in C; Otherwise, K is irreducible. When we remove all reducible elements from C, the new irreducible covering is called reduct of C and denoted by reduct(C).*

3 Covering-Based Rough Set Model for User-Based Collaborative Filtering

3.1 Purpose

For user-based CF, if fewer users in the top of a similarity list are selected as neighbors of an active user, high-accuracy items could be recommended for the active user; however, the types of recommendations will be decreased, even in just making the most popular items as recommendations. If more types of items are to be recommended, more users should be selected as neighbors of the active user, but the accuracy will decrease as the number of neighbors grows. Therefore, it is difficult for CF to simultaneously obtain good values for metrics of accuracy and coverage. To solve this problem, the relative effective neighbors should be selected from all neighbors such that the recommendations not only maintain good values of accuracy but also obtain satisfactory values of coverage.

3.2 Model Constructions

In this subsection, we present detailed information and the steps comprising CBCF, which does not use any user demographic data. In short, CBCF needs the following information:

The users set U: $U = \{u_1, u_2 ... u_E\}$, where E is the number of users.

The items set S: $S = \{s_1, s_2 ... s_I\}$, where I is the number of items.

The rating function f: $U \times S \rightarrow R$, $r_{x,i} = f(x, i)$ represents the rating score of user x for item i. Here,

$$r_{x,i} = \begin{cases} \gamma, & \text{the rating score,} \\ \star, & \text{no rating score.} \end{cases}$$

Furthermore, θ is set as the threshold for rating score, and items with $r_{x,i} \geq \theta$ are defined as items relevant to user x.

The item's attributes set A: $A = \{\alpha_1, \alpha_2 ... \alpha_P\}$, where α_n is an attribute of the item and P is the number of attributes.

Input: — The query of the active user in the following form:

$$[\alpha_1 = v_1] \wedge [\alpha_2 = v_2] \wedge ... \wedge [\alpha_m = v_m],$$

where v_m is a value of α_m.

Output: — A set of recommended items $Rec \subset S$

Step 1: Set $Rec = \emptyset$.

Step 2: Use the rating information and cosine-based similarity approach to compute the similarity between the active user and remaining users. The top K % of users in the similarity list, which are defined as L, are selected as neighbors of the active user. Here, we have

$$sim(x, y) = \frac{\sum_{i \in S_{xy}} r_{x,i} r_{y,i}}{\sqrt{\sum_{i \in S_{xy}} r_{x,i}^2} \sqrt{\sum_{i \in S_{xy}} r_{y,i}^2}}, \tag{1}$$

where $sim(x, y)$ indicates the similarity between users x and y and S_{xy} is the set of all items rated by both users x and y.

Step 3: The decision class X consists of all items that fit the active user's query options. Otherwise, some options will be rejected as minimally as possible until an adequate decision class X is obtained.

Step 4: Setting decision class X as the domain, and for each neighbor $j \in L$, relevant items of the neighbor j in domain X are a common covering C_j, where

$$C_j = \{i \in X | r_{j,i} \geq \theta\}. \tag{2}$$

Here, we define $C* = X - \cup C_j$; then, $C = \{C_1, C_2 ... C_n, C*\}$ is a covering for the active user in domain X. If the set $\cup C_j$ is empty, the domain is enlarged from X to S.

Step 5: On the basis of covering reduction in the covering-based rough set, redundant neighbors are removed from covering C of the active user to obtain reduct(C). The active user's reduct-neighbors, which are defined as L^*, consist of all users in reduct(C), and the relevant items of L^* comprise the candidates D for the active user.

Step 6: Depending on the ratings of L^*, scores are predicted for D using the adjusted weighted sum approach, i.e.,

$$P_{x,i} = \bar{r}_x + k \sum_{y \in U^*} sim(x,y) * (r_{y,i} - \bar{r}_y), \tag{3}$$

where U^* represents the neighbors in L^* who have rated item i. Then, the average rating $\bar{r}_x$ of user x is defined as

$$\bar{r}_x = \frac{\sum_{i \in S_x} r_{x,i}}{card(S_x)}, where \quad S_x = \{i \in S | r_{x,i} \neq \star\}.$$

In the above, $P_{x,i}$ is the prediction of item i for user x, and multiplier k serves as a normalizing factor and is selected as

$$k = \frac{1}{\sum_{y \in U^*} sim(x,y)}. \tag{4}$$

When all predictions are completed, the top N items in the prediction list defined as D_N are selected as the recommended items.

Step 7: Set $Rec = D_N$; output Rec.

3.3 Model Discussion

Reduction of covering is a core component of CBCF, which applies the notion of covering reduction to select relative effective neighbors. To remove redundant neighbors to the extent possible, decision class X is defined as the domain of the active user; therefore, the covering of the active user can be small enough. Based on the definition of reduction in covering-based rough set theory, for common covering C_i, if there exists another common covering C_j for which $C_i \subset C_j$, C_i will be considered as reducible and therefore removable. In this model, C_i denotes the relevant items of neighbor i, and $C_i \subset C_j$ indicates that neighbor j has more relevant items than neighbor i. In other words, neighbor j will be more efficient than neighbor i in domain X for making recommendations; therefore, neighbor i can be removed. Removing all reducible common coverings means that all relative effective neighbors are selected from all neighbors such that this model could just use the relative effective neighbors to make recommendations.

4 Experiments and Evaluation

4.1 Experimental Setup and Evaluation Metrics

For our experiments, we used the MovieLens [4] popular dataset, as it has often been utilized to evaluate RSs. The ratings dataset consists of 1682 movies, 943 users and a total of 100,000 ratings on a scale of 1 to 5. Each user has rated at least 20 movies, and for our study, movies rated above 3 were treated as the user's relevant movies. Furthermore, the covering reduction algorithm defined by Definition 3 was used for our experiments.

We also used the conventional leave-one-out procedure to evaluate the performance of our model. Items that the active user has rated were treated as unrated items, and our model predicted a rating score for every unrated item using information obtained from the remaining users. We summed every attribute's value in the relevant movies dataset, and two attributes with the largest sums are selected as query options of the active user.

To measure the performance of our new model, we use mean absolute error (MAE), root mean square error (RMSE), coverage, precision, recall, and F1 as evaluation metrics, all of which are popular metrics for evaluating RSs. Moreover, the reduction rate is defined as an evaluation metric, which measures the capability of removing redundant neighbors from all neighbors and is given as

$$ReductionRate = \frac{1}{card(U)} \sum_{u \in U} \frac{card(L_u - L_u^*)}{card(L_u)}, \tag{5}$$

where L_u denotes neighbors of the user u, and L_u^* means the reduct-neighbors of user u.

4.2 Comparing CBCF with the Un-reduction Model

We further define Un-CBCF to represent the model without the use of covering reduction. In experiments of both CBCF and Un-CBCF, the top 50% users of the similarity list were selected as the active user's neighbors. To obtain precision, recall, and F1, the number of recommendations were set as 2, 4, 6, 8, 10, and 12.

Table 1. Results of evaluation metrics between CBCF and Un-CBCF

	MAE	RMSE	Reduction Rate	Coverage
CBCF	0.681	0.853	0.795	81.002
Un-CBCF	0.658	0.817	-	88.929

Table 1 illustrates the results of our evaluation metrics between CBCF and Un-CBCF, with values of MAE, RMSE and coverage of Un-CBCF being slightly better than CBCF. For the reduction rate, which only applies to CBCF, on the average, approximately 79.5% of neighbors are removed as redundant neighbors. Given that there are 943 users in the MovieLens dataset and the top 50% users of the similarity list were selected as neighbors, the number of neighbors for Un-CBCF was 471, whereas after making the reduction, the average number of reduct-neighbors for CBCF was only 97.

Figures 1, 2, and 3 show the precision, recall, and F1 metrics, respectively. As shown in the figure, precision had high values for both CBCF and Un-CBCF. The recall and F1 values became higher as the number of recommendations grew. Overall, the precision, recall, and F1 values were almost the same between CBCF and Un-CBCF.

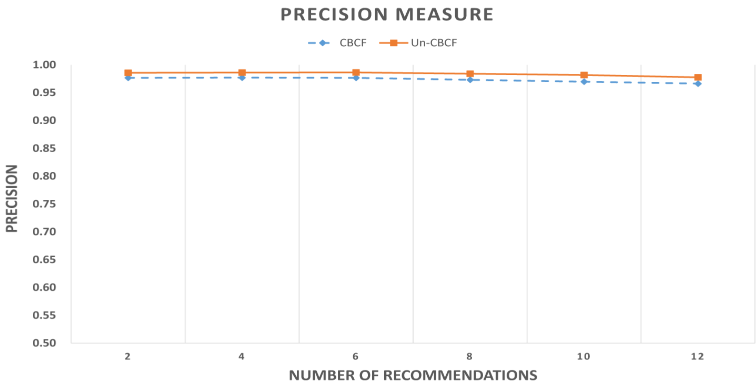

Fig. 1. Precision for the CBCF and Un-CBCF versus the number of recommendations

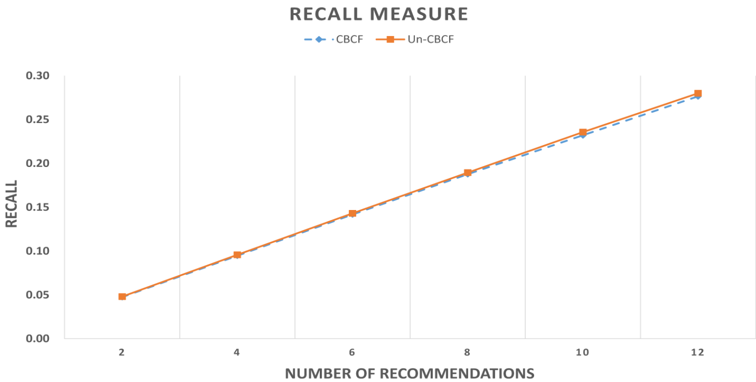

Fig. 2. Recall for the CBCF and Un-CBCF versus the number of recommendations

From these comparative results for CBCF and Un-CBCF, we conclude that our new CBCF model required an average of 97 reduct-neighbors to obtain almost the same results as the Un-CBCF model, which used 471 neighbors; therefore, our new CBCF model would be much more efficient in a RS.

4.3 Comparing CBCF with the Classic CF Model

To further illustrate the performance of our new model, we compared results with the classic CF approach. From the results of reduction in CBCF, on the

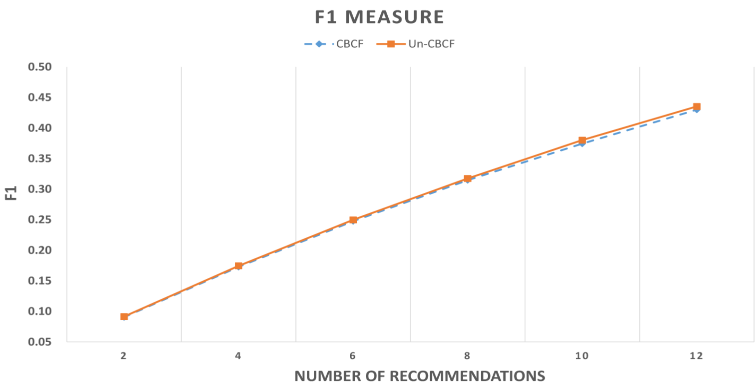

Fig. 3. F1 for the CBCF and Un-CBCF versus the number of recommendations

average, 97 reduct-neighbors were used to make prediction, so in this experiment, the top 97 users of the similarity list were selected as the active user's neighbors for CF.

Table 2. Results of evaluation metrics between CBCF and CF

	MAE	RMSE	Coverage
CBCF	0.681	0.853	81.002
CF	0.675	0.851	50.026

Table 2 shows the results of evaluation metrics for CBCF and CF. As illustrated in the table, values of MAE and RMSE for CF were satisfactory, meaning that the predicted scores were close to the original scores; however, coverage was not good enough, indicating that CF recommended 50% of the items which the active user had not rated. For our new CBCF model, coverage was improved to 81% with the values of MAE and RMSE satisfying. Figures 4, 5, and 6 illustrate the precision, recall, and F1 measures for CBCF versus CF, respectively. From the figures, we note that CBCF was better than CF in terms of precision, decreasing as the number of recommendation grew; however, the values of recall and F1 increased as the number of recommendations grew. Overall, all values were almost the same between CBCF and CF.

Comparative results between CBCF and CF reveal that our new CBCF model could overcome the disadvantage of CF and obtain sufficiently good values of coverage and accuracy.

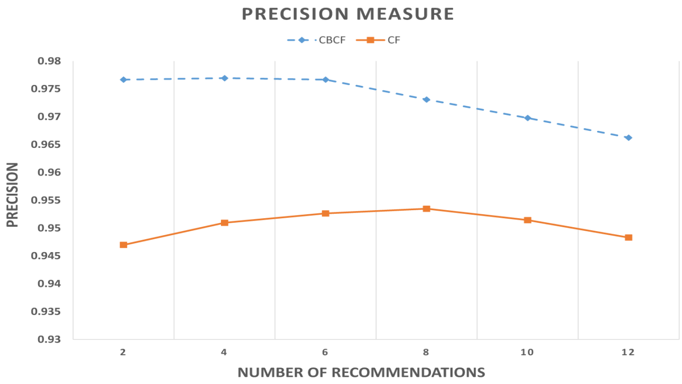

Fig. 4. Precision for CBCF and CF versus the number of recommendations

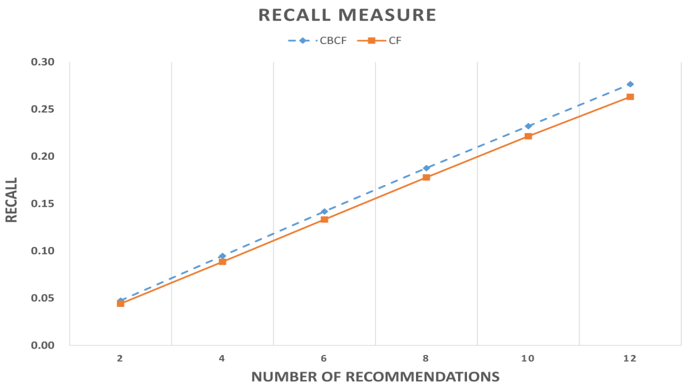

Fig. 5. Recall for CBCF and CF versus the number of recommendations

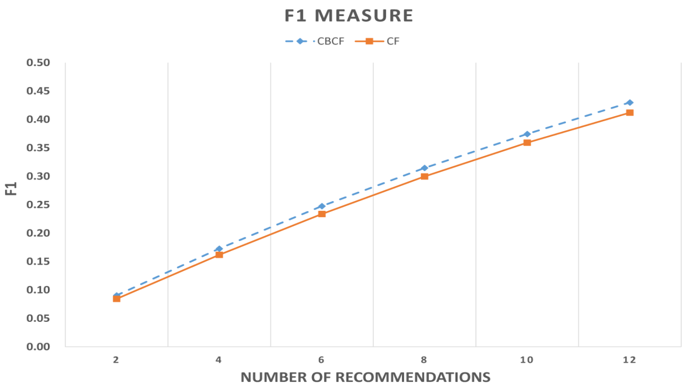

Fig. 6. F1 for CBCF and CF versus the number of recommendations

5 Conclusions and Future Work

CF is the most commonly used and studied technology for making recommendations in RSs. User-based CF has been the most popular recommendation method to date. Generally, we use accuracy and coverage to evaluate the RS, but for user-based CF, if we obtain a high level of accuracy, coverage tends to be unsatisfactory, meaning the RS could only recommend a small set of items. On the contrary, if coverage is outstanding, accuracy tends to decrease, causing the RS to recommend items inexactly, thus causing users to potentially stop using the RS. It is difficult for user-based CF to simultaneously provide satisfying accuracy and coverage.

Therefore, in this study, we presented a new CBCF model based on covering-based rough sets to improve user-based CF, which treats relevant items of every neighbor as a common covering for the active user, and then utilizes covering reduction theory to remove redundant neighbors. The reduct-neighbors are used to predict the score for each candidate, and our approach selects the top N candidates as recommendations. Results of our experiments illustrate that CBCF was able to utilize fewer neighbors to produce almost the same results compared with the model using all neighbors. Moreover, unlike user-based CF, CBCF was able to simultaneously obtain satisfactory accuracy and coverage; therefore, CBCF could be used in practice in the effective operation of RSs.

In our future work, we plan to use upper and lower approximation operators of a covering in covering-based rough set to generate candidates for a RS, because the upper and lower approximation operations are the same between reduction and un-reduction for a covering. We can only utilize reduct-neighbors to obtain the same candidates generated by upper and lower approximation operations, so that accuracy and coverage of the model could be more satisfactory.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering **17**, 734–749 (2005)
2. Bobadilla, J., Ortega, F.: Recommender System Survey. Knowledge-Based Systems **46**, 109–132 (2013)
3. Herlocker, J.L., Konstan, J.A.: An Empirical Analysis of Design Choices in Neighborhood-based Collaborative Filtering Algorithms. Information Retrieval **5**, 287–310 (2002)
4. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 230–237. ACM Press, New York (1999)
5. Pawlak, Z.: Rough Sets. International Journal of Computer and Information Sciences **11**, 341–356 (1982)
6. Pomykala, J.A.: Approximation Operations in Approximation Space. Bull. Pol. Acad. Sci. **35**, 653–662 (1987)

7. Su, X., Khoshgoftaar, T.M.: A Survey of Collaborative Filtering Techniques. Advance in Artificial Intelligence **2009**, 1–19 (2009)
8. Symeonidis, P., Nanopoulos, A., Papadopoulos, A.N., Manolopoulos, Y.: Collaborative Recommender Systems: Combining Effectiveness and Efficiency. Expert Systems with Application **34**, 2995–3013 (2008)
9. Tsang, E., Cheng, D., Lee, J., Yeung, D.: On the upper approximations of covering generalized rough sets. In: Proceedings of the 3rd International Conference Machine Learning and Cybernetics, pp. 4200–4203. IEEE Press, Shanghai (2004)
10. Wang, J., Dai, D., Zhou, Z.: Fuzzy Covering Generalized Rough Sets. Journal of Zhoukou Teachers College **21**, 20–22 (2004)
11. Yang, T.: Li.Q.G.: Reduction about Approximation Spaces of Covering Generalized Rough Sets. International Journal of Approximate Reasoning **51**, 335–345 (2010)
12. Zakowski, W.: Approximations in the Space (u, π). Demonstration Math. **16**, 761–769 (1983)
13. Zhu, W.: Relationship among Basic Concepts in Covering-based Rough Sets. Information Sciences **179**, 2478–2486 (2009)
14. Zhu, W.: Relationship between Generalized Rough Sets Based on Binary Relation. Information Sciences **179**, 210–225 (2009)
15. Zhu, W.: Topological Approached to Covering Rough Sets. Information Sciences **177**, 1499–1508 (2007)
16. Zhu, W., Wang, F.Y.: The Fourth Type of Covering-based Rough Sets. Information Sciences **201**, 80–92 (2012)
17. Zhu, W., Wang, F.Y.: Reduction and Maximization of Covering Generalized Rough Sets. Information Sciences **152**, 217–230 (2003)
18. Zhu, W., Wang, F.Y.: On Three Types of Covering-Based Rough Sets. IEEE Transactions on Knowledge and Data Engineering **19**, 1131–1144 (2007)