Guidelines for Article Preparation for the Transactions on Rough Sets

James F. Peters[1] and Andrzej Skowron[2]

[1] Department of Electrical and Computer Engineering, University of Manitoba
Winnipeg, Manitoba R3T 5V6, Canada
jfpeters@ee.umanitoba.ca
[2] Institute of Mathematics, Warsaw University
Banacha 2, 02-097 Warsaw, Poland
skowron@mimuw.edu.pl

Abstract. This document[3] focuses on how to prepare a research article for submission to the *Transactions on Rough Sets* (TRS), which is a journal subline of the *Lecture Notes in Computer Science* (LNCS) published by Springer[4]. The TRS journal is an outgrowth of the pioneering work on rough sets by Zdzisław Pawlak and many others during the past 25 years. This journal is the primary publication of the International Rough Set Society (IRSS). These Guidelines present the requirements, in general, for the style and structure for TRS articles[5]. In particular, this document illustrates each of the parts (*e.g.*, abstract, keywords, introduction, organization of sections, notation, algorithms, theory, formulas, figures, tables, conclusion, acknowledgements and references) of a typical TRS article. In addition, there are also instructions for those authors of TRS articles whose native language is not English. LaTeX and the Springer *llncs* document class are to be used to prepare an article for submission to the TRS. Details about how to obtain and use LaTeX are also given in these Guidelines. The principal contribution of these Guidelines is an integrated approach to the preparation of research papers for the TRS.

Keywords: Article, guidelines, IRRS, LaTeX, LNCS, rough sets, style, TRS.

1 Introduction

The *Transactions on Rough Sets* (*TRS* [41])[6] journal is an outgrowth of the pioneering work on rough sets by Zdzisław Pawlak (see, *e.g.*, [21–26]) and many others [31] during the past three decades. The TRS is the premier publication for the International Rough Set Society (IRSS [8]). All papers accepted for inclusion in the TRS will be published by Springer as a journal subline of the *Lecture Notes*

[3] Revised 14 November 2006 (see [8] for details).
[4] See http://www.springer.com/west/home/computer/lncs?SGWID=4-164-6-99627-0
[5] The LaTeX sources used to create these Guidelines are available at [8].

in Computer Science (LNCS). All papers accepted for inclusion in the TRS will be published by Springer as a journal subline of the *Lecture Notes in Computer Science* (LNCS). Hence, there are very strict guidelines for preparation of high quality research papers that will be submitted to the TRS. Needless to say, any paper that does not adhere to these guidelines will be returned to the authors of the paper without any review.

The TRS journal has as its principal aim the fostering of professional exchanges between scientists and practitioners who are interested in the foundations and applications of rough sets. Topics include foundations and applications of rough sets as well as foundations and applications of hybrid methods combining rough sets with other approaches important for the development of intelligent systems. The journal includes high-quality research articles accepted for publication on the basis of thorough peer reviews. Dissertations and monographs up to 250 pages formatted in the usual style of TRS articles that include new research results can also be considered as regular papers. Extended and revised versions of selected papers from conferences can also be included in regular or special issues of the journal.

This Style Guideline for preparing an article for submission to the TRS has the following organization. The LaTeX document class to be used by every author is specified in Sect. 2. In addition, information about LaTeX as well as information about public domain software for LaTeX users is also given in Sect. 2. A very detailed set of style guidelines that must be used in preparing an article for the TRS can be found in Sect. 3. Notice that this document illustrates how each of the article components described in the Style Guideline can be performed in LaTeX. In effect, this document can be used as a template for a TRS article. The method of submission of an article to the TRS is given in Sect. 4. Finally, the specification for the concluding section of a TRS article is given in Sect. 5.

2 LaTeX Requirements

Prepare your TRS paper using **LaTeX** and the Springer-Verlag document class llncs.cls (see [35]). There are a number of books that are helpful in showing how to use LaTeX (see, *e.g.*, [7, 5, 17]). A tutorial to help you get started in preparing a LaTeX document is available at [13]. To find the latest version of LaTeX, try [14]. Articles written in Microsoft ® Word will not be accepted.

MiKTeX (pronounced mik-tech) is an up-to-date implementation of TeX [16], which is available for different platforms[7]. For example, TeX and related programs is available for Windows on a PC (all current variants), MacOS X, Linux, and Unix. TeX is a typesetting system invented by D.E. Knuth [10], who has written that LaTeX is "intended for the creation of beautiful books–and especially for books that contain a lot mathematics". More information about TeX (its history, novel aspects of TeX, examples, software, notes, references as well as external links) can be found at [38]. The MiKTeX system is public domain software that is available from [15]. The `graphicx`, `algorithm2e`, and `subfigure`

[7] See $http://www.tex.ac.uk/cgi-bin/texfaq2html?label=TeXsystems$.

packages used in this document, are among the files that can be downloaded from the web (see, *e.g.*, [1, 6, 36]).

It is also important to have access to a LaTeX editor such as TeXnicCenter, which is an Integrated Development Environment for writing TeX/LaTeX-documents on Microsoft Windows ® platforms[40]. TeXnicCenter provides a lot of enhanced features like structure parsing, project management, syntax highlighting, and so on. TeXnicCenter is also public domain software available from [39].

Symbols in LaTeX

A comprehensive list of 3300 symbols commonly available to LaTeX users as well as the corresponding LaTeX commands used to produce the symbols is available from [12]. Tables 1-3 present some of the more commonly used symbols in LaTeX.

Table 1. Lowercase Greek Letters

α	\alpha	θ	\theta	o	\o	τ	\tau
β	\beta	ϑ	\vartheta	π	\pi	υ	\upsilon
γ	\gamma	ι	\iota	ϖ	\varpi	ϕ	\phi
δ	\delta	κ	\kappa	ρ	\rho	φ	\varphi
ϵ	\epsilon	λ	\lambda	ϱ	\varho	χ	\chi
ε	\varepsilon	μ	\mu	σ	\sigma	ψ	\psi
ζ	\zeta	ν	\nu	ς	\varsigma	ω	\omega
η	\eta	ξ	\xi				

Table 2. Uppercase Greek Letters

Γ	\Gamma	Λ	\Lambda	Σ	\Sigma	Ψ	\Psi
Δ	\Delta	Ξ	\Xi	Υ	\Upsilon	Ω	\Omega
Θ	\Theta	Π	\Pi	Φ	\Phi		

Table 3. Relation Symbols

$\in$	\in	$\notin$	\notin	$\cap$	\cap	$\cup$	\cup
$\approx$	\approx	$\rightarrow$	\rightarrow	$\prec$	\prec	$\subset$	\subset
$\asymp$	\asymp	$\bowtie$	\Join	$\preceq$	\preceq	$\subseteq$	\subseteq
$\bowtie$	\bowtie	$\leq$	\leq	$\propto$	\proto	$\succ$	\succ
$\cong$	\cong	$\ll$	\ll	$\sim$	\sim	$\succeq$	\succeq
$\dashv$	\dashv	$\mid$	\mid	$\simeq$	\simeq	$\supset$	\supset
$\doteq$	\doteq	$\models$	\models	$\smile$	\smile	$\supseteq$	\supseteq
$\equiv$	\equiv	$\neq$	\neq	$\vdash$	\vdash	$\forall$	\forall
$\gg$	\gg	$\ni$	\ni	$\frown$	\frown	$\perp$	\perp
$\geq$	\geq	$\parallel$	\parallel	$\exists$	\exists	$\emptyset$	\emptyset

3 Structure of a TRS Article

Papers submitted to the TRS should have the structure described in this Section.

3.1 Email Addresses

The email address should be included at the end of the postal address for each author, *not* just for the first author of a submitted paper. See, for example, the style used for authors, addresses and email addresses used at the beginning of this Guideline.

3.2 Abstract

The abstract for your article should briefly describe the problem(s) solved in your article. Include in your description, a brief indication of the context for your research. Then indicate if you have included such things as examples, sample experimental results, and data. Finally, clearly state the contribution of your article. Abstracts are usually between 100 and 150 words in length (longer abstracts are also acceptable).

3.3 Keywords

A *keyword* is a significant or descriptive word that helps identify a focal point in the subject matter (*i.e.*, principal topic) of a research article. Keywords provide a basis for searches for published articles on a particular subject, and have been the subject of considerable research (see, *e.g.*, [42]). After your Abstract in a TRS article, give a list of up to 8 keywords or phrases that indicate the principal topics covered in your article.

3.4 Introduction

The Transactions on Rough Sets (TRS) is the primary publication of the International Rough Set Society (IRSS [8]). Your article should begin with a thorough introduction. This Introduction should begin by clearly describing the problem(s) to be solved in your article. The description of the problem should be followed by an indication of the approach to solving the problem in your article. The problem-description should include a carefully crafted overview of the context for the problem. This context must include references to the principal (and, possibly, minor) works that are currently published on this subject (see, *e.g.*, [31] for a comprehensive list of papers on rough sets and their applications). After giving a description of the problem(s) solved in your article, clearly indicate the contribution(s) of your article. In doing this, you should compare and contrast your contribution(s) with related contributions in papers published by others. Finally, your introduction should conclude with an overview of the organization of your article.

3.5 Basic Concepts

After your introduction, introduce a section that gives an overview of the basic concepts that underly the contribution(s) of your article. This Basic Concepts

section should be sufficiently detailed so that the connection between traditional basic concepts and notation and the new results presented by you in succeeding sections of your article, is made clear. Document the sources of the concepts and notation that you introduce in this Basic Concepts section.

3.6 Details

Include sections (and subsections) that give further details about the research introduced in the article. Such sections should make it clear what is intended by the research presented in the article. This can be accomplished by first giving a detailed presentation of notation, data, experiments, algorithms, and theory related to what you present. After that, compare and contrast the results and methods presented in your article with results and methods presented by other researchers. This means that you must document the claims made in your article, and give a complete presentation of results found in works related to your research. By doing this, you will provide the reader with a clearcut view of your research and its context.

3.7 Notation

As an aid to understanding the intricacies of the research contribution presented in your article, be sure to introduce, explain, and document the notation that you use. It is preferable to use notation traditionally employed in rough set theory rather than using new notation that is equivalent to the common notation. In other words, the following rule-of-thumb should be followed: *be conservative in the introduction of new notation in connection with your research.* Otherwise, it becomes very difficult for reviewers to follow what you are writing. If you find it necessary to introduce new notation, then it should be clearly explained, illustrated, and distinguished from existing notation.

3.8 Data

Include sections (and subsections) that give a detailed description of the data used to obtain experimental results. This description should include sample data and an indication of the sources of your data. It will be very good if the data used in your article is made available either at [8] or at some other website.

3.9 Experiments

Include sections (and subsections) that give a detailed presentation of experimental results used in support of the claims made in your paper. It is recommended that a presentation of the experimental results include carefully labeled plots, charts, and tables. Document your experimental results by comparing and contrasting them to results obtained by other researchers.

Algorithm 1: Off-Policy Monte Carlo Control Algorithm

Input : States $s \in S$, Actions $a \in A(s)$ //$A(s)$ is a set of actions in state s.
Output: Policy $\pi(s)$ //where $\pi(s)$ is a policy in state s that controls the selection of a particular action in state s.

for (*all* $s \in S, a \in A(s)$) **do**
  $\pi(s)$ is randomly chosen;
  $Q(s,a) \longleftarrow$ arbitrary; //where Q is the value of an action a is state s
  $N(s,a) \longleftarrow 0$; //numerator of $Q(s,a)$
  $D(s,a) \longleftarrow 0$; //denominator of $Q(s,a)$
end
while *True* **do**
  Select an action policy $\pi'(s,a)$, and use it to generate an episode:

  $s_0, a_0, r_1, s_1, a_1, r_2, \ldots, s_{T-1}, a_{T-1}, r_T, s_T$; // r_i is the reward on action a_{i-1}

  $\tau \longleftarrow$ the latest time at which $a_\tau \neq \pi(s_\tau)$;
  for *each pair* (s,a) *appearing in the episode at time* τ *or later* **do**
    $t \longleftarrow$ the time of first occurrence of s, a such that $t \geq \tau$;
    $w \longleftarrow \prod_{k=t+1}^{T-1} \frac{1}{\pi'(s_k, a_k)}$; // weight on $R_t = \sum_{i=t+1}^{T} r_i$;
    $N(s,a) \longleftarrow N(s,a) + wR_t$;
    $D(s,a) \longleftarrow D(s,a) + w$;
    $Q(s,a) \longleftarrow \frac{N(s,a)}{D(s,a)}$;
  end
  for *each* $s \in S$ **do**
    $\pi(s) \longleftarrow argmax_a Q(s,a)$;
  end
end

3.10 Algorithms

Include detailed algorithms that specify the steps used in the research presented in the article. Each algorithm should be accompanied by an explanation of the principal steps, and the significance of the notation used in the algorithm. Document your algorithms by comparing and contrasting them with methods used by other researchers. Consider, for example, the reinforcement learning algorithm given in Alg. 1, which uses the `algorithm2e` package. This algorithm represents a form of off-policy Monte Carlo control (general case), which is explained in detail in [28, 37].

Your algorithm should begin by specifying the input needed to perform the steps of the algorithm, and the output produced by the algorithm. Notice that Alg. 1 includes a specification of the input (pre-condition) and output (post-condition) for the algorithm. To specify your algorithm as shown in Alg. 1, you need to use the `algorithm2e` package. A very detailed description of the `algorithm2e` package with complete, detailed examples is available at [1].

Your algorithm should begin by specifying the input needed to perform the steps of the algorithm as well as the output produced by the algorithm. Notice that the LaTeX for Alg. 1 specifies the algorithm input as follows:

```
\Input{States $s\in \textsl{S} $,~Actions~$ a\in A(s) $.}
```

and the output produced by Alg. 1 is specified as follows:

```
\Output{Policy $\pi(s)$.}
```

3.11 Theory

In the case where a new theory or theoretical results concerning rough sets is introduced, then sections (and subsections) should be introduced that present the notation, formal definitions, facts, lemmas, propositions and/or theorems that underly the new theory. Proofs of properties, principal lemmas, propositions and theorems should be included as part of the presentation of the underpinnings of the new theory. Document your theory by comparing and contrasting your definitions, lemmas, propositions and theorems with theories introduced by other researchers. By way of illustration of a formal definition as well as a theorem and its proof, consider the following brief example about a well-known property of the empty set.

Definition 1. Truth Table for Implication. If P and S are statements that can be only *true* or *false*, then the truth table for implication (denoted by $\Rightarrow$) is shown in the table 4.

Table 4. Truth table for implication

P	S	$P \Rightarrow S$
true	*true*	*true*
true	*false*	*false*
false	*false*	*true*
false	*true*	*true*

Definition 2. Equality of sets. Let $\forall x \,.\, x \in A$ denote *for all x such that x belongs to an arbitrary universe A*[8]. The sets A and B are equal if and only if (1) and (2) hold true.

$$\forall x \,.\, x \in A \;\Rightarrow\; x \in B. \tag{1}$$

$$\forall x \,.\, x \in B \;\Rightarrow\; x \in A. \tag{2}$$

Definition 3. Empty Set The *empty set* is a set which has no elements.

In other words, there exists no element that belongs to the empty set, or for each element of any universe of objects, a given element does not belong to the empty set, which is expressed by (3).

$$\forall x \,.\, x \notin \emptyset. \tag{3}$$

[8] The notation $\forall$ (called a *universal quantifier*) was introduced by Gerhard Gentzen in 1935 [4]. The dot notation '.' comes from [33], and serves as a means of eliminating parentheses in an assertion [30].

Theorem 1. *(The uniqueness of the null set) There is exactly one empty set.*

Proof. Let sets A and B be sets satisfying the definition of the empty set. From Def. 3, we obtain (4) and (5).

$$\forall x \; . \; x \notin A. \tag{4}$$

$$\forall x \; . \; x \notin B. \tag{5}$$

From (4), the premise in (6) is false, and from (5), the statement in the conclusion in (6) is false. Hence, from the truth table in Def. 1, the implication in (6) is true. Similarly, the implication in (7) is true.

$$\forall x \; . \; x \in A \;\Rightarrow\; x \in B. \tag{6}$$

$$\forall x \; . \; x \in B \;\Rightarrow\; x \in A. \tag{7}$$

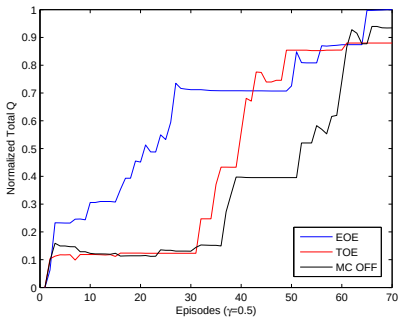
Thus, sets A and B satisfy the requirements for the the equality of sets (see Def. 2) and, therefore, $A = B$, which proves that there is exactly one empty set.

■

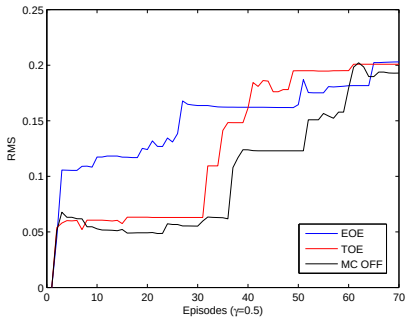
3.12 Formulas

All formulas should be numbered, and properly punctuated and formatted. Every equation should end with a period "." unless there is a continuation which is marked by a comma ",". Please punctuate a displayed equation in the same way as ordinary text, but with a small space in front of the end punctuation. Consider, for example, the problem of determining the value of an action a in state s with a function $Q^{\pi}(s, a)$, where a policy π is followed. During the episodic behavior of a system that learns from experience, $Q^{\pi}(s, a)$ is defined relative to the expected value of returns. The return $R = r_1 + \gamma r_2 + \gamma^2 r_3 + ... + \gamma^{T-1} r_T$ (*i.e.*, cumulative future discounted rewards) results from a sequence of actions, where $\gamma \in [0, 1]$ is called a discount rate and r_i is the i^{ih} reward. The basic idea is to choose actions during an episode that ends in a terminal state at time T so that the expected discounted return $E^{\pi}(R)$ following policy π improves. A *policy* $\pi(s, a)$ is a mapping from an environment state s to the probability of selecting a particular action. In choosing actions, it is necessary to estimate the expected value of R. Let $Pr(X = x)$ denote the probability that X equals x. It is assumed that the return R (cumulative discounted future rewards) for a sequence of actions is a discrete random variable, and the probability $Pr(R = r)$ is not known. In effect, if the episodic behavior of a swarm yields a sequence of returns $R_1, \ldots, R_n$ over n episodes, the value of the expectation $E[R] = \sum_{j=1}^{n} x_j Pr(R_j = x_j)$ is not known. Monte Carlo methods (see, *e.g.*, [32]) offer an approach to estimating the expected value of R. Briefly, using the Monte Carlo method, $Q^{\pi}(s, a)$ can be estimated using a weighted sum. Let w_i denote an importance sampling weight on R_i, and we obtain an approximate value of $Q^{\pi}(s, a)$ using Eq. 8.

$$Q^{\pi}(s, a) \approx \frac{\sum_{i=1}^{n} w_i R_i}{\sum_{i=1}^{n} w_i}, \tag{8}$$

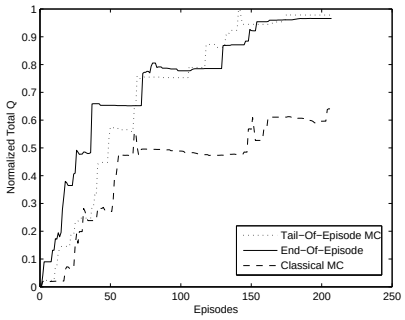


1.1: Normalized Total $Q(s, a)$/episode

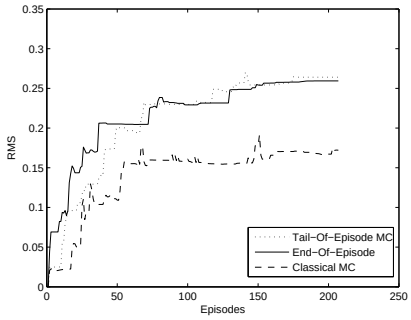


1.2: RMS/ episode

Fig. 1. Off-Policy Monte Carlo Control Methods



2.1: Normalized Total $Q(s, a)$/episode



2.2: RMS/ episode

Fig. 2. Off-Policy MC Greyscale Plots

where $\approx$ denotes *approximately equal to*. It is sometimes necessary to define a function with multiple lines. In that case, put a comma "," at the end of each line before the last line, and put a "." at the end of the last line. For example, standard rough inclusion(SRI) ν_{SRI} can be defined for any X, $Y \subseteq U$ as in Eq. 9.

$$\nu_{SRI}(X, Y) = \begin{cases} \frac{|X \cap Y|}{|X|}, if X \neq \emptyset, \\ 1, \quad if X = \emptyset. \end{cases} \tag{9}$$

In other words, $\nu_{SRI}(X, Y)$ returns the degree of overlap between X and Y. In the case where $X = Y$, then $\nu_{SRI}(X, Y) = 1$. Under the assumption that $X \neq \emptyset$, the minimum inclusion value $\nu_{SRI}(X, Y) = 0$, is obtained when $X \cap Y = \emptyset$ (*i.e.*, X and Y have no elements in common). There are two rules-of-thumb to observe relative to formulas. First, after you give a formula, give an explanation of each of the parts of the formula for the reader (aka *reviewer of your article*).

Second, numbered formulas serve both as an aid in reading as well as writing a research article. Hence, whenever an article contains many formulas, numbering of the formulas is highly recommended.

3.13 Figures

Each figure should be accompanied by a short, meaningful caption. Either before or after a figure is given, give an explanation of the parts of the figure shown. For example, the results of reinforcement learning by a swarm inside an ecosystem is shown in Fig. 1. The plot of normalized total Q values during 70 reinforcement learning episodes is shown in Fig. 1.1 and RMS per episode for both the conventional, end-of-episode (EOE) and tail-of-episode (TOE) off-policy Monte Carlo algorithms is shown in Fig. 1.2. Even though the EOE algorithm does better than the conventional off-policy and TOE algorithms, the difference between the values in the plots in Fig. 1.1 is not sufficient to conclude that one algorithm is better than the other. The oscillations in the two plots shown in Fig. 1.2 for both algorithms is not surprising, when one considers how Q values are estimated (see, *e.g.*, [37, 28]). Since Figures are usually not printed in color in the TRS, it is then necessary to convert color pictures to greyscale (see, *e.g.*, Fig. 2).

3.14 Images for Figures.

eps files are commonly used to supply images in Figures in a LaTeX article. A public domain utility for converting .jpg image files to .eps files can be downloaded from [9]. Included in the zip file for use with MS DOS is jpeg2ps.exe written by Thomas Merz. jpeg2ps can be used to convert JPEG files to PostScript Level 2 or 3 EPS. Merz writes that "jpeg2ps is not really a converter but a wrapper: it reads the image parameters (width, height, number of color components) in a JPEG file, writes the corresponding EPS header and then copies the compressed JPEG data to the output file". To see how to convert jpg to eps files, put the jpeg2ps.exe file in the directory where you are working. Also put a jpg file you wish to convert to eps format, in your working directory. Open the DOS Command Prompt window in your working directory. Then, for example, the command line to use to convert EOFF_Q_G05.jpg to an eps file is shown in Fig. 3.

3.15 Conclusion

The concluding section of your article should summarize the contributions of your article relative to what you have presented. You should comment on the significance of your results. In addition, you should indicate possible extensions of what you have presented, and planned future work.

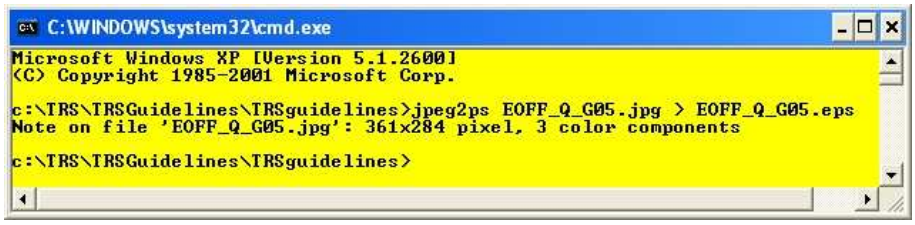


Fig. 3. Sample jpg To eps Conversion

3.16 References

A separate section entitled *References* should be included in your article. It is required that all references are proper and complete. The format of references should follow the guidelines contained in Springer's guide for LNCS authors [35]. The use of bibTEX is encouraged, especially in case of long reference lists. The authors should apply the bibliography (`splncs.bst`) style from Springer. It is very important that you follow the style for TRS references as illustrated in the following list.

- **Article in journal:** see, *e.g.*, [3, 21–24, 26, 28, 42],
- **Article in proceedings:** see, *e.g.*, [34],
- **Article in book:** see, *e.g.*, [11, 18, 27],
- **Book:** see, *e.g.*, [5, 7, 17, 19, 20, 29, 32, 37],
- **Research Report:** see, *e.g.*, [25, 2],
- **Website:** see, *e.g.*, [1, 8–10, 15, 16, 31, 35, 39, 41, 43].

3.17 Required Declaration About Article

Prepare a letter containing a statement that indicates that your article has not been published or submitted to any other local or international conference or journal. This means that your article should not have published or submitted in English or any other language to any other publication. All author(s) of the article must sign this letter. Please send the signed letter by fax (204 261 4639) to Professor James F. Peters, Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, R3T 5V6, Canada.

3.18 English

Before you submit your article to the TRS, you must have someone whose native language is English proof-read and, if necessary, correct the English in your article in the case where English is not your native language.

3.19 Copyright

Extensions of articles published in a conference or workshop proceedings not published and copyrighted by Springer will *not* be accepted for publication in the TRS. The exception to this rule is the following: permission from a publisher must be obtained for any part of an article that has been published in a non-Springer conference or workshop proceedings that is used in the article you have submitted to the TRS. Permission from the publisher to include specific items (diagrams, figures, tables, paragraphs) must be clearly indicated in an article submitted to a TRS special issue (do this by introducing a footnote having the form shown[9]). In addition, a letter from the publisher granting permission for each copyrighted item to appear in your submitted TRS article must be forwarded to the TRS Editors-in-Chief listed in Table 4.

In the case where an article submitted to a TRS special issue is an extension of a previous article, the author(s) of the submitted article must do the following: (1) give references to the earlier, published paper(s), and (2) indicate precisely how the submitted article differs from the earlier publication(s).

4 Submission Method

All papers submitted to the TRS must be in pdf format. To submit a completed paper, send the pdf file containing your paper to each of the Editors-in-Chief listed in Table 5[10].

Table 5. TRS Editors-in-Chief

James F. Peters	jfpeters@ee.umanitoba.ca
Andrzej Skowron	skowron@mimuw.edu.pl

Upon acceptance, the authors will be requested to provide all LaTeX source files, pdf file, images (.eps files) and any additional files used to prepare an article. The TRS editorial office must be able to compile a submitted article without encountering errors in the LaTeX source files or in the article structure.

5 Conclusion

The last section of your paper contains concluding remarks concerning what you have presented in the article prepared for submission to the TRS. You should concisely state what you have done in your article. Then highlight the main contribution of your article. Finally, briefly state the future work that you envision as a result of the research described in your paper.

[9] *Name of the publisher* and a *declaration* that permission to publish the specific item in the article submitted to the TRS has been granted

[10] Notice that the caption must always be placed **above** the table.

Acknowledgements

The authors extend their thanks to Maciej Borkowski, Jerzy Grzymała-Busse, David Gunderson,Christopher Henry, Dan Lockery, Sheela Ramanna and Marcin Szczuka for suggestions concerning the LaTeX used in these Guidelines. The research of James F. Peters and Andrzej Skowron is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) grant 185986 and grant 3 T11C 002 26 from Ministry of Scientific Research and Information Technology of the Republic of Poland, respectively.

Note: Acknowledgements should appear in an unnumbered section immediately after the Conclusion section. In addition, to mentioning the names of those persons and/or organizations that have been helpful in writing your article. If appropriate, it is common to mention the sources of support of the research presented in the article. For example, you might write "This research has been supported by (enter the name of funding agency plus grant number(s))".

References

1. Algorithm2e: See documentation (includes sample algorithms) at `www.tug.org/tex-archive/macros/latex/contrib/algorithm2e/algorithm2e.pdf`
2. Borkowski, M., Peters, J.F.: *Approximation Space Approach to Matching Image Segments*. Research Report, Computational Intelligence Laboratory, Department of Electrical & Computer Engineering, University of Manitoba (2005). See, also, Borkowski, M., Peters, J.F. (2006) in vol. V of the *Transactions on Rough Sets journal* [3].
3. Borkowski, M., Peters, J.F.: Matching 2D image segments with genetic algorithms and approximation spaces, *Transactions on Rough Sets journal* V (2006) 63-102.
4. Gentzen, G.: Untersuchen über das logische Schliessen I. *Mathematische Zeitschrift* 39 (1935) 176-210.
   See `http://dz-srv1.sub.uni-goettingen.de/cache/toc/D17147.html`
5. Goosens, M., Rahtz, S., Mittelbach: The LaTeX Graphics Companion. Illustrating Documents with TeX and Postscript. Addison Wesley Longman, Inc., Harlow, England (1997).
6. graphicx package at `http://www.tug.org/tex-archive/macros/latex/contrib/graphicx-psmin/`.
7. Grätzer, G.: Math into LaTeX, 3rd Edition. Birkhäuser, Boston, MA, U.S.A., c/o Springer-Verlag, New York (2004).
8. International Rough Set Society: `http://roughsets.home.pl/www/`
9. jpeg2ps utility for converting jpg to eps file available at `http://gnuwin32.sourceforge.net/packages/jpeg2ps.htm`
10. Knuth, D.E.: Message to all users of TeX at `http://www-cs-faculty.stanford.edu/~knuth/cm.html`
11. Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A.: Rough sets: A tutorial. In: [20] (1999) 3–98.

12. LaTeX symbols and corresponding LaTeX commands that produce the symbols[11]: http://www.ctan.org/tex-archive/info/symbols/comprehensive/
13. LaTeX tutorial to help you get started: http://lahelper.sourceforge.net/mini_latex_tutorial.html
14. For the latest version and current developments with LaTeX, see http://www.latex-project.org/
15. MikTeX implementation of TeXfor the Windows operating system: http://www.miktex.org/
16. MikTeX 2.4 Manual. Revision 2.4.1779, 7 June 2006: http://docs.miktex.org/manual/
17. Mittelbach, F., Goossens, M.: *The LaTeX Companion*, 2nd Edition. Addison-Wesley, Boston, MA, U.S.A. (2004).
18. Pal, S.K., Peters, J.F., Polkowski, L., Skowron, A.: Rough-neural computing: An introduction. In: [19] (2004) 15–42.
19. Pal, S.K., Polkowski, L., Skowron, A. (Eds.): *Rough-Neural Computing. Techniques for Computing with Words*. Springer-Verlag, Berlin, Germany (2004).
20. Pal, S.K., Skowron, A. (Eds.): *Rough-Fuzzy Hybridization: A New Trend in Decision-Making*. Springer-Verlag, Singapore Pte. Ltd. (1999).
21. Peters, J.F., Skowron, A.:, Zdzisław Pawlak life and work (1906-2006), *Information Sciences* 177 (2007) 1-2.
22. Pawlak, Z., Skowron, A.: Rudiments of rough sets, *Information Sciences* 177 (2007) 3-27.
23. Z. Pawlak, A. Skowron, Rough sets: Some extensions, *Information Sciences* 177 (2007) 28-40.
24. Z. Pawlak, A. Skowron, Rough sets and Boolean reasoning, *Information Sciences* 177 (2007) 41-73.
25. Pawlak Z.: *Classification of Objects by Means of Attributes*, Institute for Computer Science, Polish Academy of Sciences, Report 429, March (1981).
26. Pawlak, Z.: Rough sets. *International Journal of Computer and Information Science* 11 (1982) 341–356.
27. Pawlak, Z., Skowron, A.: Rough membership functions. In: Yager, R., et al. (Eds.), *Advances in Dempster Shafer Theory of Evidence*. Wiley, N.Y., U.S.A. (1994) 251–271.
28. Peters, J.F., Henry, C.: Reinforcement learning with approximation spaces. *Fundamenta Informaticae* 71 (2006) 323–349.
29. Polkowski, L.: *Rough Sets. Mathematical Foundations*. Physica–Verlag, Berlin, Germany (2002).
30. Quine, W.V.O.: Mathematical Logic. Harper & Row, Publishers, NY (1951).
31. Rough Set Database System (RSDS): http://rsds.wsiz.rzeszow.pl/rsds.php
32. Rubinstein, R.Y.: *Simulation and the Monte Carlo Method*. John Wiley & Sons, Toronto, Canada (1981).
33. Russell, A.N., Whitehead, A.N.: Principia Mathematica, 3 vols. Cambridge University Press, London, GB, 2nd Ed. (1927).
34. Skowron, A., Stepaniuk, J.: Generalized approximation spaces. In: *Proceedings of the Third International Workshop on Rough Sets and Soft Computing*, San Jose, CA, U.S.A. (1994) 156–163.
35. Springer LNCS: See LaTeX files at http://www.springer.com/east/home/computer/lncs?SGWID=5-164-7-72376-0

[11] **Note**: some symbols require font files available from http://www.ctan.org.

36. subfigure package examples at
http://www.hep.man.ac.uk/u/jenny/jcwdocs/latex/figures.html
37. Sutton, R.S., Barto, A.G.: *Reinforcement Learning: An Introduction*. The MIT Press, Cambridge, MA, U.S.A. (1998).
38. For detailed information about TEX, see
http://en.wikipedia.org/wiki/TeX
39. TeXnicCenter system: http://sourceforge.net/projects/texniccenter/
40. TeXnicCenter system features, tools, add-ons, downloads:
http://www.texniccenter.org/
41. Transactions on Rough Sets (TRS):
http://www.springer.com/west/home/computer/lncs?SGWID=
4-164-6-99627-0(see also http://roughsets.home.pl/www/ for Guidelines for Special Issues of the TRS).
42. Turney, P.D.: Learning algorithms for keyphrase extraction. *Information Retrieval* 2/4 (2000) 303-336.
43. WITAS project (2001): http://www.ida.liu.se/ext/witas/eng.html