COMPUTATIONAL INTELLIGENCE

INTRODUCTION

There are a number of interpretations of the notion of Computational Intelligence (CI) (1–9). Computationally intelligent systems have been characterized by Bezdek (1, 2) relative to adaptivity, fault-tolerance, speed, and error rates. In its original conception, a number of technologies were identified to constitute the backbone of Computational Intelligence, namely, neural networks (75, 76), genetic algorithms (75, 76), fuzzy sets and fuzzy systems (75, 76), evolutionary programming (75, 76) and artificial life (10, 11). More recently, rough set theory and its extensions to approximate reasoning and real-time decision systems have been considered in the context of computationally intelligent systems (3,6–9,12,13,46,75,76), which naturally led to the generalization along the line of Granular Computing. Overall, CI can be regarded as a field of intelligent system design and analysis which dwells upon a well-defined and clearly manifested synergy of genetic, granular and neural computing. A detailed introduction to the different facets of such a synergy along with a discussion of various realizations of such synergistic links between CI technologies is given in (3,4,44,46,65,66,75,76).

GENETIC ALGORITHMS

Genetic algorithms were proposed by Holland as a search mechanism in artificially adaptive populations (14). A *genetic algorithm* (*GA*) is a problem-solving method that simulates Darwinian evolutionary processes and naturally occurring genetic operations on chromosomes (15). In nature, a *chromosome* is a threadlike linear strand of DNA and associated proteins in the nucleus of animal and plant cells. A chromosome carries genes and serves as a vehicle in transmitting hereditary information. A *gene* is a hereditary unit which occupies a specific location on a chromosome and which determines a particular trait in an organism. Genes can undergo mutation (alteration or structural change). A consequence of the mutation of genes is the creation of a new trait in an organism. In genetic algorithms, the traits of artificial life forms are stored in bit strings which mimic chromosome strings found in nature. The traits of individuals in a population are represented by a set of evolving chromosomes. A GA transforms a set of chromosomes to obtain the next generation of an evolving population. Such transformations are the result of applying operations such as reproduction based on survival of the fittest and genetic operations such as sexual recombination (also called crossover) and mutation.

Each artificial chromosome has an associated fitness, which is measured with a fitness function. The simplest form of fitness function is known as raw fitness, which is some form of performance score (e.g., number of pieces of food found, amount of energy consumed, number of other life forms found). Each chromosome is assigned a probability of reproduction which is proportional to its fitness. In a Darwinian system, natural selection controls evolution (16). Consider, for example, a collection of artificial life forms with behaviors resembling ants. Fitness will be measured relative to the total number of pieces of food found and eaten (partially eaten food is counted). Reproduction consists in selecting the fittest individual x and weakest individual y in a population, and replacing y with a copy of x. After reproduction, a population will then have two copies of the fittest individual. A crossover operation consists in exchanging genetic coding (bit values of one or more genes) in two different chromosomes. The steps in a crossover operation are (1) randomly select a location (also called interstitial location) between two bits in a chromosome string to form two fragments, (2) select two parents (chromosomes to be crossed), and (3) interchange the chromosome fragments. Because of the complexity of traits represented by a gene, substrings of bits in a chromosome are used to represent a trait (17). The evolution of a population resulting from the application of genetic operations results in changing fitness of individual population members. A principal goal of GAs is to derive a population with optimal fitness.

The pioneering works of Holland (15) and L. J. Fogel and others (18) gave birth to the new paradigm of population-driven computing (evolutionary computation) resulting in structural and parametric optimization. Evolutionary programming was introduced by L. J. Fogel in the 1960s (19). The evolution of competing algorithms defines *evolutionary programming*. Each algorithm operates on a sequence of symbols to produce an output symbol that is likely to maximize an algorithm's performance relative to a well-defined payoff function. Evolutionary programming is the precursor of genetic programming (15). In *genetic programming*, large populations of computer programs are genetically bred.

FUZZY SETS AND SYSTEMS

A fuzzy systems (models) are immediate constructs that results from a description of real-world systems (say, social, economic, ecological, engineering, or biological) in terms of information granules- fuzzy sets and relationships between them (20). The concept of fuzzy set introduced by Zadeh in 1965 (21, 22) becomes of paramount relevance when formalizing a notion of partial membership of element. Fuzzy sets are distinguished from the fundamental notion of a set (also called a crisp set) by the fact that their boundaries are formed by elements with whose degree of belongingness are allowed to assume numeric values in the interval [0, 1]. Let us recall that the characteristic function for a set X returns a Boolean value {0, 1} indicating whether an element x is in X or is excluded from it. A fuzzy set is non-crisp inasmuch as the characteristic function for a fuzzy set returns a value in [0, 1]. Let U, X, $\tilde{A}$, x be a universe of objects, subset of U, fuzzy set in U, and an individual object x in X, respectively. For a set X, $\mu_{\tilde{A}}: X \rightarrow [0, 1]$ is a function which determines the degree of membership of an object x in X. A fuzzy set $\tilde{A}$ is then defined to be a set of ordered pairs where $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \, \varepsilon \, X\}$. The counterparts of intersection and union (crisp sets) are the t-norm and s-norm operators in fuzzy set theory. For the intersection of fuzzy sets, the min operator was suggested by Zadeh (29), and belongs to a class of intersection operators (min, product, bold intersection) known as triangular or t-norms. A t-norm is a mapping $t : [0, 1]^2 \rightarrow [0, 1]$. The s-norm (t-conorm) is a mapping $s : [0, 1]^2 \rightarrow [0, 1]$ (also triangular co-norm) is commonly used for the union of fuzzy sets. The properties of triangular norms are presented in (82).

Fuzzy sets exploit imprecision in conventional systems in an attempt to make system complexity manageable. It has been observed that fuzzy set theory offers a new model of vagueness (13). Many examples of fuzzy systems are given in Pedrycz (23), and in Kruse, Gebhardt, and Klawonn (24).

NEURAL COMPUTING

Neural networks offer a powerful and distributed computing architecture equipped with significant learning abilities (predominantly as far as parametric learning is concerned). They help represent highly nonlinear and multivariable relationships between system variables. Starting from pioneering research of McCulloch and Pitts (25), and others (26, 27), neural networks have undergone a significant metamorphosis and have become an important reservoir of various learning methods (28) as well as an extension of conventional techniques in statistical pattern recognition (29). Artificial Neural Networks (ANNs) were introduced to model features of the human nervous system (25). An *artificial neural network* is collection of highly interconnected processing elements called neurons. In ANNs, a neuron is a threshold device, which aggregates ("sums") its weighted inputs, and applies an activation function to each aggregation to produce a response. The summing part of a neuron in an ANN is called an Adaptive Linear Combiner (ALC) in (30, 31). A McCulloch-Pitts neuron n_i is a binary threshold unit with an ALC that computes a weighted sum net where net = $\sum_{j=0}^{n} w_j x_j$. A weight w_i associated with x_i represents the strength of connection of the input to a neuron. Input x_0 represents a bias, which can thought of as an input with weight 1. The response of a neuron can be computed in a number of ways. For example, the response of neuron n_i can be computed using sgn(*net*), where sgn(*net*) = 1 for $net > 0$, sgn(*net*) = 0 for $net = 0$, and sgn(*net*) = −1, if $net < 0$. A neuron comes with adaptive capabilities that could be fully exploited assuming that there is an effective procedure is introduced to modify the strengths of connections so that a correct response is obtained for a given input. A good discussion of learning algorithms for various forms of neural networks can be found in Freeman and Skapura (32) and Bishop (29). Various forms of neural networks have been successfully used in system modeling, pattern recognition, robotics, and process control applications (46,50,51,54,75,76).

ROUGH SETS

Rough sets introduced by Pawlak in 1981 (77, 78) and elaborated in (13,33,34,67,68,74,79–81) offer another approach to CI by drawing attention to the importance of set approximation in knowledge discovery and information granulation. Rough set theory also offers a model for approximation of vague concepts (69, 83).

In particular, rough set methods provide a means of approximating a set by other sets (33, 34). For computational reasons, a syntactic representation of knowledge is provided by rough sets in the form of data tables. In general, an information system IS is represented by a pair (U, F), where U is a non-empty set of objects and F is a non-empty, countable set of probe functions that are a source of measurements associated with object features. For example, a feature of an image may be color with probe functions that measure tristimulus values received from three primary color sensors, brightness (luminous flux), hue (dominant wavelength in a mixture of light waves), and saturation (amount of white light mixed with a hue). Each $f \varepsilon F$ maps an object to some value. In effect, we have $f : U \rightarrow V_f$ for every $f \varepsilon F$.

The notions of equivalence and equivalence class are fundamental in rough sets theory. A binary relation $R \subseteqq X \times X$ is an equivalence relation if it is reflexive, symmetric and transitive. A relation R is reflexive if every object $x \varepsilon X$ has relation R to itself. That is, we can assert $x R x$. The symmetric property holds for relation R if xRy implies yRx for every $x, y \varepsilon X$. The relation R is transitive for every $x, y, z \varepsilon X$, then xRy and yRz imply xRz. The equivalence class of an object $x \varepsilon X$ consists of all objects $y \varepsilon X$ so that xRy. For each $B \subseteqq A$, there is associated an equivalence relation $Ind_A(B) = \{(x, x') \mid \forall\alpha \varepsilon B.\ \alpha(x) = \alpha(x')\}$ (indiscernibility relation). If $(x, x') \varepsilon Ind_A(B)$, we say that objects x and x' are indiscernible from each other relative to attributes from B. This is a fundamental concept in rough sets. The notation $[x]_B$ is a commonly used shorthand that denotes the equivalence class defined by x relative to a feature set B. In effect, $[x]_B = \{y \varepsilon U \mid x\ Ind_A(B)\ y\}$. Further, partition $U/Ind_A(B)$ denotes the family of all equivalence classes of relation $Ind_A(B)$ on U. Equivalence classes of the indiscernibility relation (called B-granules generated by the set of features B (13)) represent granules of an elementary portion of knowledge we are able to perceive relative to available data. Such a view of knowledge has led to the study of concept approximation (40) and pattern extraction (41). For $X \varepsilon U$, the set X can be approximated only from information contained in B by constructing a B-lower and B-upper approximation denoted by $B_*X = \{x \varepsilon U \mid [x]_B \subseteqq X\}$ and $B^*X = \{x \varepsilon U \mid [x]_B \cap X \neq \emptyset\}$, respectively. In other words, a lower approximation B_*X of a set X is a collection of objects that can be classified with full certainty as members of X using the knowledge represented by features in B. By contrast, an upper approximation B^*X of a set X is a collection of objects representing both certain and possible uncertain knowledge. In the case where B_*X is a proper subset of B^*X, then the objects in X cannot be classified with certainty, and the set X is rough. It has recently been observed by Pawlak (13) that this is exactly the idea of vagueness proposed by Frege (41). That is, the vagueness of a set stems from its borderline region.

The size of the difference between lower and upper approximations of a set (i.e., boundary region) provides a basis for the "roughness" of an approximation. This is important because vagueness is allocated to some regions of what is known as the universe of discourse (space) rather than to the whole space as encountered in fuzzy sets. The study of what it means *to be a part of* provides a basis for what is known as mereology introduced by Lesniewski in 1927 (36). More recently, the study of what it means to be a part of *to a degree* has led to a calculus of granules (8,37–39,71,73). In effect, granular computing allows us to quantify uncertainty and take advantage of uncertainty rather than blindly discarding it.

Approximation spaces introduced by Pawlak (77), elaborated by (33,34,66,69,70–73), applied in (6–8,40,46,59,64) serve as a formal counterpart of our perception ability or observation (69), and provide a framework for approximate reasoning about vague concepts. In its simplest form, an approximation space is any pair (U, R), where U is a nonempty set of objects (called a universe of discourse) and R is an equivalence relation on U (called an indiscernibililty relation). Equivalence classes of an indiscernibility relation are called elementary sets (or information granules) determined by R. Given an approximation space $S = (U, R)$, a subset X of U is definable if it can be represented as the union of some of the elementary sets determined by R. It was originally observed that not all subsets of U are definable in S (69). Given a non-definable subset X of U, our observation restricted by R causes X to be perceived as a vague object. An upper approximation B^*X is the least definable subset of U containing X, and the lower approximation B_*X is the greatest definable subset of U contained in X.

Fuzzy set theory and rough set theory taken singly and in combination pave the way for a variety of approximate reasoning systems and applications representing a synergy of technologies from computational intelligence. This synergy can be found, for example, in recent work on the relation between fuzzy sets and rough sets (13,35,46,60,65), rough mereology (37–39,65,66), rough control (42, 43), fuzzy-rough-evolutionary control (44), machine learning (34,45,59), fuzzy neurocomputing (3), rough neurocomputing (46) diagnostic systems (34, 47), multi-agent systems (8,9,48), real-time decision-making (12, 49), robotics and unmanned vehicles (50–53), signal analysis (55), and software engineering (4,55–58).

BIBLIOGRAPHY

1. J. C. Bezdek, On the relationship between neural networks, pattern recognition and intelligence, *Int. J. Approximate Reasoning*, **6**, 1992, 85–107.
2. J. C. Bezdek, What is Computational Intelligence? In: J. Zurada, R. Marks, C. Robinson (Eds.), *Computational Intelligence: Imitating Life, Piscataway*, IEEE Press, 1994, 1–12.
3. W. Pedrycz, *Computational Intelligence: An Introduction*. Boca Raton, FL: CRC Press, 1998.
4. W. Pedrycz, J. F. Peters (Eds.), *Computational Intelligence in Software Engineering, Advances in Fuzzy Systems—Applications and Theory*, vol. **16**. Singapore: World Scientific, 1998.
5. D. Poole, A. Mackworth, R. Goebel, *Computational Intelligence: A Logical Approach*. Oxford: Oxford University Press, 1998.
6. N. Cercone, A. Skowron, N. Zhong (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing Special Issue. *Computation Intelligence: An International Journal*, vol. **17**, no. 3, 2001, 399–603.
7. A. Skowron, S. K. Pal (Eds.), Rough Sets, Pattern Recognition and Data Mining Special Issue. *Pattern Recognition Letters*, vol. **24**, no. 6, 2003, 829–933.
8. A. Skowron, Toward intelligent systems: Calculi of information granules. In: T. Terano, T. Nishida, A. Namatane, S. Tsumoto, Y. Ohsawa, T. Washio (Eds.), New Frontiers in Artificial Intelligence, *Lecture Notes in Artificial Intelligence 2253*. Berlin: Springer-Verlag, 2001, 28–39.
9. J. F. Peters, A. Skowron, J. Stepaniuk, S. Ramanna, Towards an ontology of approximate reason, *Fundamenta Informaticae*, vol. **51**, nos. 1–2, 2002, 157–173.
10. R. Marks, Intelligence: Computational versus Artificial, *IEEE Trans. on Neural Networks*, **4**, 1993, 737–739.
11. D. Fogel, Review of“ Computational Intelligence: Imitating Life”, *IEEE Trans. on Neural Networks*, **6**, 1995, 1562–1565.
12. J. F. Peters, Time and Clock Information Systems: Concepts and Roughly Fuzzy Petri Net Models. In: J. Kacprzyk (Ed.), *Knowledge Discovery and Rough Sets*. Berlin: Physica Verlag, a division of Springer Verlag, 1998.
13. Z. Pawlak, A. Skowron, Rudiments of rough sets, *Information Sciences*, **177**, 2006, 3–27. See, also, J. F. Peters, A. Skowron, Zdzislaw Pawlak life and work (1926–2006), *Information Sciences*, **177**, 1–2, Z. Pawlak, A. Skowron, Rough sets: Some extensions, *Information Sciences*, **177**, 28–40 and Z. Pawlak, A. Skowron, Rough sets and Boolean reasoning, *Information Sciences*, **177**, 41–73.
14. J. H. Holland, Adaptive plans optimal for payoff-only environments, *Proc. of the Second Hawaii Int. Conf. on System Sciences*, 1969, 917–920.
15. J. R. Koza, *Genetic Programming: On the Progamming of Computers by Means of Natural Selection*. Cambridge, MA: MIT Press, 1993.
16. C. Darwin, On the Origin of Species by Means of Natural Selection, or *the Preservation of Favoured Races in the Struggle for Life*. London: John Murray, 1959.
17. L. Chambers, *Practical Handbook of Genetic Algorithms*, vol. **I**. Boca Raton, FL: CRC Press, 1995.
18. L. J. Fogel, A. J. Owens, M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, Chichester, J. Wiley, 1966.
19. L. J. Fogel, *On the organization of the intellect*. Ph.D. diss., UCLA, 1964.
20. R. R. Yager and D. P. Filev, *Essentials of Fuzzy Modeling and Control*. NY: John Wiley & Sons, Inc., 1994.
21. L. A. Zadeh, Fuzzy sets, *Information and Control*, **8**, 1965, 338–353.
22. L. A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans. on Systems, Man, and Cybernetics*, **2**, 1973, 28–44.
23. W. Pedrycz, *Fuzzy Control and Fuzzy Systems*, NY: John Wiley & Sons, Inc., 1993.
24. R. Kruse, J. Gebhardt, F. Klawonn, *Foundations of Fuzzy Systems*. NY: John Wiley & Sons, Inc., 1994.
25. W. S. McCulloch, W. Pitts, A logical calculus of ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics* **5**, 1943, 115–133.
26. F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Washington: Spartan Press, 1961.
27. M. Minsky, S. Pappert, *Perceptrons: An Introduction to Computational Geometry*, Cambridge: MIT Press, 1969.

28. E. Fiesler, R. Beale (Eds.), *Handbook on Neural Computation*. UK: Institute of Physics Publishing and Oxford University Press, 1997.
29. C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford: Oxford University Press, 1995.
30. B. Widrow, M. E. Hoff, Adaptive switching circuits, *Proc. IRE WESCON Convention Record*, Part 4, 1960, 96–104.
31. B. Widrow, Generalization and information storage in networks of adaline “neurons”. In M. C. Yovits, G. T. Jacobi, G. D. Goldstein (Eds.), *Self-Organizing Systems*. Washington, Spartan, 1962.
32. J. A. Freeman and D. M. Skapura, *Neural Networks: Algorithms, Applications and Programming Techniques*. Reading, MA, Addison-Wesley, 1991.
33. Z. Pawlak, Rough sets, *Int. J. of Information and Computer Sciences*, vol. **11**, no. 5, 1982, 341–356, 1982
34. Z. Pawlak, Rough Sets. *Theoretical Aspects of Reasoning about Data*, Dordrecht, Kluwer Academic Publishers, 1991.
35. W. Pedrycz, Shadowed sets: Representing and processing fuzzy sets, *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, **28**/1, Feb. 1998, 103–108.
36. S. Lesniewski, O podstawach matematyki (in Polish), *Przeglad Filozoficzny*, vol. **30**, 164–206, vol. **31**, 261–291, vol. **32**, 60–101, and vol. **33**, 142–170, 1927.
37. L. Polkowski and A. Skowron, Implementing fuzzy containment via rough inclusions: Rough mereological approach to distributed problem solving, *Proc. Fifth IEEE Int. Conf. on Fuzzy Systems*, vol. 2, New Orleans, Sept. 8–11, 1996, 1147–1153.
38. L. Polkowski, A. Skowron, Rough mereology: A new paradigm for approximate reasoning, *International Journal of Approximate Reasoning*, vol. **15**, no. 4, 1996, 333–365.
39. L. Polkowski, A. Skowron, Rough mereological calculi of granules: A rough set approach to computation, *Computational Intelligence: An International Journal*, vol. **17**, no. 3, 2001, 472–492.
40. Bazan, H. S. Nguyen, A. Skowron, M. Szczuka: A view on rough set concept approximation, In: G. Wang, Q. Liu, Y. Y. Yao, A. Skowron, *Proceedings of the Ninth International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing RSFDGrC'2003)*, Chongqing, China, 2003, LNAI 2639, 181–188.
41. J. Bazan, H. S. Nguyen, J. F. Peters, A. Skowron, M. Szczuka: Rough set approach to pattern extraction from classifiers, *Proceedings of the Workshop on Rough Sets in Knowledge Discovery and Soft Computing at ETAPS'2003*, April 12–13, 2–3, Warsaw University, electronic version in Electronic Notes in Computer Science, Elsevier, 20–29. G. Frege, *Grundlagen der Arithmetik*, **2**, Verlag von Herman Pohle, Jena, 1893.
42. T. Munakata, Z. Pawlak, Rough control: Application of rough set theory to control, *Proc. Eur. Congr. Fuzzy Intell. Technol. EUFIT'96*, 1996, 209–218.
43. J. F. Peters, A. Skowron, Z. Suraj, An application of rough set methods to automatic concurrent control design, *Fundamenta Informaticae*, **43**(1–4), 2000, 269–290.
44. T. Y. Lin, Fuzzy controllers: An integrated approach based on fuzzy logic, rough sets, and evolutionary computing. In: T. Y. Lin, N. Cercone (Eds.), *Rough Sets and Data Mining: Analysis for Imprecise Data*. Norwell, MA, Kluwer Academic Publishers, 1997, 109–122.
45. J. Grzymala-Busse, S. Y. Sedelow, W. A. Sedelow, Machine learning & knowledge acquisition, rough sets, and the English semantic code. In: T. Y. Lin, N. Cercone (Eds.), *Rough Sets and Data Mining: Analysis for Imprecise Data*. Norwell, MA, Kluwer Academic Publishers, 1997, 91–108.
46. S. K. Pal, L. Polkowski, A. Skowron (Eds.), *Rough-Neuro Computing: Techniques for Computing with Words*. Berlin: Springer-Verlag, 2003.
47. R. Hashemi, B. Pearce, R. Arani, W. Hinson, M. Paule, A fusion of rough sets, modified rough sets, and genetic algorithms for hybrid diagnostic systems. In: T. Y. Lin, N. Cercone (Eds.), *Rough Sets and Data Mining: Analysis for Imprecise Data*. Norwell, MA, Kluwer Academic Publishers, 1997, 149–176.
48. R. Ras, Resolving queries through cooperation in multi-agent systems. InT. Y. Lin, N. Cercone (Eds.), *Rough Sets and Data Mining: Analysis for Imprecise Data*. Norwell, MA, Kluwer Academic Publishers, 1997, 239–258.
49. A. Skowron, Z. Suraj, A parallel algorithm for real-time decision making: a rough set approach. *J. of Intelligent Systems*, vol. **7**, 1996, 5–28.
50. J. F. Peters, T. C. Ahn, M. Borkowski, V. Degtyaryov, S. Ramanna, Line-crawling robot navigation: A rough neurocomputing approach. In:C. Zhou, D. Maravall, D. Ruan (Eds.), *Autonomous Robotic Systems*. Berlin: Physica-Verlag, 2003, 141–164.
51. J. F. Peters, T. C. Ahn, M. Borkowski, Object-classification by a line-crawling robot: A rough neurocomputing approach. In:J. J. Alpigini, J. F. Peters, A. Skowron, N. Zhong (Eds.), *Rough Sets and Current Trends in Computing*, LNAI 2475. Springer-Verlag, Berlin, 2002, 595–601.
52. M. S. Szczuka, N. H. Son, Analysis of image sequences for unmanned aerial vehicles. In:M. Inuiguchi, S. Hirano, S. Tsumoto (Eds.), *Rough Set Theory and Granular Computing*. Berlin: Springer-Verlag, 2003, 291–300.
53. H. S. Son, A. Skowron, M. Szczuka, Situation identification by unmanned aerial vehicle. In: *Proc. of CS&P 2000*, Informatik Berichte, Humboldt-Universitat zu Berlin, 2000, 177–188.
54. J. F. Peters, L. Han, S. Ramanna, Rough neural computing in signal analysis, *Computational Intelligence*, vol. **1**, no. 3, 2001, 493–513.
55. J. F. Peters, S. Ramanna, Towards a software change classification system: A rough set approach, *Software Quality Journal*, vol. **11**, no. 2, 2003, 87–120.
56. M. Reformat, W. Pedrycz, N. J. Pizzi, Software quality analysis with the use of computational intelligence, *Information and Software Technology*, **45**, 2003, 405–417.
57. J. F. Peters, S. Ramanna, A rough sets approach to assessing software quality: Concepts and rough Petri net models. In:S. K. Pal andA. Skowron (Eds.), *Rough-Fuzzy Hybridization: New Trends in Decision Making*. Berlin: Springer-Verlag, 1999, 349–380.
58. W. Pedrycz, L. Han, J. F. Peters, S. Ramanna, R. Zhai, Calibration of software quality: Fuzzy neural and rough neural approaches. *Neurocomputing*, vol. **36**, 2001, 149–170.
59. J. F. Peters, C. Henry, Reinforcement learning with approximation spaces. *Fundamenta Informaticae*, **71** (2-3), 2006, 323–349.
60. W. Pedrycz, Granular computing with shadowed sets. In:D. Slezak, G. Wang, M. Szczuka, I. Duntsch, Y. Yao (Eds.),Rough Sets, *Fuzzy Sets, Data Mining, and Granular Computing*, LNAI 3641. Springer, Berlin 2005, 23–31.
61. W. Pedryz, Granular computing in knowledge integration and reuse. In:D. Zhang, T. M. Khoshgoftaar, M.-L. Shyu (Eds.), *IEEE Int. Conf. on Information Reuse and Integration*. Las Vegas, NV, USA, 15–17 Aug. 2005.

62. W. Pedrycz, G. Succi, Genetic granular classifiers in modeling software quality. *Journal of Systems and Software*, **76**(3), 2005, 277–285.

63. W. Pedrycz, M. Reformat, Genetically optimized logic models. *Fuzzy Sets & Systems*, **150**(2), 2005, 351–371.

64. J. F. Peters, Rough ethology: Towards a biologically-inspired study of collective behavior in intelligent systems with approximation spaces. *Transactions on Rough Sets* **III**, LNCS 3400, 2005, 153–174.

65. L. Polkowski, Rough mereology as a link between rough and fuzzy set theories: A survey. *Transactions on Rough Sets* **II**, LNCS 3135, 2004, 253–277.

66. L. Polkowski, Rough Sets. *Mathematical Foundations. Advances in Soft Computing*, Physica-Verlag, Heidelberg, 2002.

67. Z. Pawlak, Some issues on rough sets. *Transactions on Rough Sets* **I**, LNCS 3100, 2004, 1–58.

68. Z. Pawlak, A treatise on rough sets. *Transactions on Rough Sets* **IV**, LNCS 3700, 2005, 1–17.

69. E. Orlowska, *Semantics of Vague Concepts. Applications of Rough Sets*. Institute for Computer Science, Polish Academy of Sciences, Report 469, March 1981.

70. A. Skowron, J. Stepaniuk, Generalized approximation spaces. In:Lin, T. Y., Wildberger, A. M. (Eds.), *Soft Computing*, Simulation Councils, San Diego, 1995, 18–21.

71. A. Skowron, J. Stepaniuk, J. F. Peters, R. Swiniarski, Calculi of approximation spaces. *Fundamenta Informaticae*, **72** (1–3), 2006, 363–378.

72. A. Skowron, J. Stepaniuk, Tolerance approximation spaces. *Fundamenta Informaticae*, **27**(2–3), 1996, 245–253.

73. A. Skowron, R. Swiniarski, P. Synak, P., Approximation spaces and information granulation. *Transactions on Rough Sets* **III**, LNCS 3400, 2005, 175–189.

74. A. Skowron, J. F. Peters, Rough sets: Trends and Challenges. In:Wang, G.Liu, Q., Yao, Y., Skowron, A. (Eds.), *Proceedings 9th Int. Conf. on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing* (RSFDGrC2003), LNAI 2639, Springer-Verlag, Berlin, 2003, 25–34.

75. *IEEE World Congress on Computational Intelligence*. Vancouver, B.C., Canada, 16–21 July 2006.

76. Hamza, M. H. (Ed.), *Proceedings of the IASTED Int. Conf. on Computational Intelligence*. Calgary, AB, Canada, 4–6 July 2005.

77. Z. Pawlak, Classification of Objects by Means of Attributes. Institute for Computer Science, Polish Academy of Sciences, Report 429, March 1981.

78. Z. Pawlak,Rough Sets. Institute for Computer Science, Polish Academy of Sciences, Report 431, 1981.

79. Z. Pawlak, Rough classification. *Int. J. of Man-Machine Studies*, **20** (5), 1984, 127–134.

80. Z. Pawlak, Rough sets and intelligent data analysis, *Information Sciences: An International Journal*, **147** (1–4), 2002, 1–12.

81. Z. Pawlak, Rough sets, decision algorithms and Bayes' theorem, *European Journal of Operational Research*, **136**, 2002, 181–189.

82. E. P. Kement, R. Mesiar, E. Pap, *Triangular Norms*, Kluwer, Dordrecht, 2000.

83. J. Bazan, A. Skowron, R. Swiniarski, Rough sets and vague concept approximation: From sample approximation to adaptive learning, *Transactions on Rough Sets V*, LNCS 4100, Springer, Heidelberg, 2006, 39–62.

James F. Peters
Witold Pedrycz
University of Manitoba, Manitoba, Canada
University of Alberta, Alberta, Canada