

Capítulo 7

Conclusão

Este capítulo apresenta as principais conclusões deste trabalho, cujo objetivo foi propor uma extensão da FPA através da teoria fuzzy. Estão descritas suas contribuições e sugestões de trabalhos futuros.

Um dos aspectos mais positivos da técnica de Análise de Pontos por Função (FPA) é a sua simplicidade, principalmente em relação ao cálculo dos pontos por função. Essa técnica está entre as mais utilizadas para se estimar tamanho de projetos ou sistemas de software. Durante o processo de contagem dos pontos, que representam a dimensão do projeto ou da aplicação, cada função é classificada de acordo com sua complexidade funcional relativa.

Embora pontos por função representem a funcionalidade de um sistema, muitos estudos empíricos apontam para uma relação existente entre esses pontos e o esforço de trabalho necessário para desenvolvê-los (Albrecht, 1983; Kemerer, 1997). Medidas derivadas a partir de pontos por função, como custo e prazo, podem estimar valores não factíveis em consequência do modo atual de classificar a funcionalidade das funções que constituem o sistema.

Muitos estudos já propuseram estender a FPA, objetivando, principalmente, obter uma maior precisão na pontuação de sistemas de maior complexidade algorítmica, como a *Full Function Point* (FFP, 1997), utilizada para sistemas de tempo real e equipamentos com software embutido, e a métrica *Use Case Points* (Karner, 1993; Fihlman, 2000).

Este trabalho propõe a utilização de conceitos e propriedades da teoria dos conjuntos *fuzzy* para estender a FPA em FFPA (Análise de Pontos por Função *Fuzzy*). A teoria *fuzzy* busca construir uma estrutura formal quantitativa, capaz de capturar as imprecisões do conhecimento humano. Com os pontos por função produzidos através da FFPA, valores derivados como prazo e

custo de desenvolvimento podem ser obtidos com maior precisão, segundo o que foi constatado no estudo de caso realizado.

A atividade de validação modelo FFPA foi árdua, especialmente no levantamento das seguintes informações: a produtividade da equipe e as datas de identificação de início e fim de cada um dos sistemas legados governamentais avaliados, incluindo projetos de desenvolvimento e de manutenção.

A produtividade de software é um fenômeno complexo de se tratar, onde estão contabilizados pelo menos 250 fatores que podem influenciá-la (Jones, 2000). Poder-se-ia levar anos até determinar a produtividade média de uma empresa de grande porte. Embora se saiba que valores obtidos por *benchmark* podem ser tendenciosos (Kitchenham, 2001), além de não representarem fielmente a produtividade de uma empresa em particular, utilizaram-se os valores da SPR – *Software Productivity Research* (Jones, 1996), diante das dificuldades apresentadas.

Por causa das questões acima citadas, a base histórica, tida à disposição para a validação do modelo FFPA proposto, foi drasticamente reduzida a um pequeno número de projetos. Desses projetos, foram utilizados no estudo de casos dois de desenvolvimento e sete de manutenção.

Como resultados deste trabalho de extensão da FPA para FFPA, pode-se destacar:

- *através da utilização de números fuzzy trapezoidais para os termos lingüísticos baixa, média e alta, funções que se encontram nas regiões limítrofes dos intervalos utilizados passaram a receber valores com uma graduação contínua, sem uma mudança abrupta desses valores;*
- *criação do termo lingüístico de complexidade muito alta, pertencente a um intervalo parametrizado, através do valor de “k”, que pode ser ajustado, segundo as características da organização, para melhor lidar com os sistemas de grande porte.*
- *esse modelo forneceu uma estimativa de prazo de programação mais precisa que a FPA padrão, especialmente, quando foram avaliados sistemas que ultrapassaram o limite de alta complexidade, referenciando uma grande quantidade de itens de dados e arquivos em um mesmo processo elementar;*

- *o modelo torna a técnica mais sensível à modificação da funcionalidade existente, fazendo com que os pontos por função da aplicação reflita os resultados da manutenção. Isto permite um melhor gerenciamento da evolução do sistema;*
- *construção de um protótipo, que automatiza o cálculo de pontos por função usando FFPA.*

Dentre os trabalhos futuros que podem dar continuidade à pesquisa aqui apresentada, sugere-se:

- *aplicação do modelo FFPA a outras extensões da FPA como FFP e Use Case Points, visto que são as propostas de extensão mais promissoras;*
- *utilização desse modelo em outros domínios de aplicação (neste trabalho foi utilizada uma base de sistemas governamentais); e*
- *transformação do protótipo em Java em uma ferramenta mais abrangente, acrescentando-lhe a capacidade de capturar informações adicionais que possam interferir nas estimativas de prazo e custo para o desenvolvimento e manutenção de sistemas.*