

Uma Abordagem Baseada em Lógica Nebulosa para Seleção de Equipes Distribuídas Tecnicamente Qualificadas

Vinicius Souza dos Santos, Gledson Elias

COMPOSE – Component Oriented Software Engineering Group
Departamento de Informática, Universidade Federal da Paraíba

vinicius@compose.ufpb.br, gledson@di.ufpb.br

***Abstract.** The selection process of human resources is complex and inherently subjective, requiring to measure and represent knowledge and experience of professionals. Such difficulty is even higher in a distributed setting. Furthermore, in the context of software projects, specific needs should be taken into account to make possible to identify and select the best skilled professionals to develop a project. Thereby, in order to abstract complexity and subjectivity, this paper presents a fuzzy logic based approach for selecting technically skilled teams to implement software modules of distributed software product line projects.*

***Resumo.** O processo de seleção de recursos humanos é complexo e inerentemente subjetivo, exigindo mensurar e representar o conhecimento e experiência dos profissionais. A dificuldade é ainda maior em uma conjuntura distribuída. Além disso, no contexto de projetos de software, necessidades específicas devem ser levadas em consideração para tornar possível identificar e selecionar os profissionais mais adequados para desenvolver um projeto. Dessa forma, a fim de tratar a complexidade e a subjetividade, este trabalho apresenta uma abordagem baseada em lógica nebulosa (fuzzy) para seleção de equipes distribuídas tecnicamente qualificadas a implementar módulos de software de projetos de linhas de produto de software.*

1. Introdução

Com o objetivo de reduzir custos e melhorar a qualidade de software, muitas empresas estão adotando abordagens de reuso de software, como as Linhas de Produto de Software (LPS), que possibilitam a instanciação de diferentes produtos a partir de uma mesma plataforma, provendo reuso sistemático e planejado. Contudo, para se alcançar sucesso numa LPS é necessário a construção de uma plataforma consistente e abrangente, o que requer especialistas de domínio e proximidade com o mercado alvo [Clements *et al* 2002]. De maneira complementar, outra abordagem que vem sendo amplamente adotada em função da globalização é o Desenvolvimento Distribuído de Software (DDS). Dentre as principais motivações para adoção do DDS está a possibilidade de aproximação do mercado alvo e acesso a profissionais mais qualificados e especialistas no domínio [Karolak 1999].

Portanto, no cenário de desenvolvimento distribuído de linhas de produto de software, a seleção de recursos tecnicamente qualificados se torna um fator estratégico chave na gerência de um projeto de software. A seleção de equipes de desenvolvimento distribuídas é apenas uma etapa na alocação de recurso, que envolve demais aspectos

como disponibilidade, tempo e custo [Schwalbe 2002]. Ainda assim, essa atividade por si só é bastante complexa, já que pode gerar diversas soluções tornando difícil ou inviável para o gerente de projetos fazer a melhor escolha baseando-se apenas em sua experiência e intuição.

No processo de seleção, características específicas das equipes e de seus membros devem ser medidas e avaliadas, servindo como um filtro para encontrar boas soluções. O problema com esse tipo de avaliação é a subjetividade inerente à representação e medição do conhecimento. Ao representar o conhecimento como um valor booleano não se é capaz de representar níveis diferentes de conhecimento, necessários num processo de seleção. Por outro lado a representação de níveis através de números, apesar de eficiente para uma máquina, é contra-intuitivo para os humanos, pois o conhecimento não pode ser mensurado com precisão. Uma alternativa é a utilização de lógica nebulosa (*fuzzy*) que, como evidenciado por [Zadeh 1965], permite a representação de conceitos e comparações subjetivas escondendo toda a complexidade numérica, podendo assim ser usada como um mecanismo de suporte à decisão.

Assim, com o intuito de mitigar o problema da complexidade e subjetividade na seleção de equipes distribuídas tecnicamente qualificadas, este trabalho propõe uma abordagem baseada em lógica nebulosa. Considerando os requisitos técnicos dos módulos de software, as habilidades e o conhecimento das equipes candidatas, bem como políticas de seleção definidas pelo gerente de projeto, a abordagem proposta permite selecionar o conjunto de equipes tecnicamente mais adequadas para desenvolver as especificidades dos módulos de software que compõem o projeto de uma linha de produtos de software.

Vale ressaltar que a abordagem proposta representa uma das fases do framework de recomendação para alocação de equipes de desenvolvimento em projetos distribuídos de linhas de produtos de software, cujo conjunto de fases foi brevemente introduzido em [Santos *et al* 2010]. Assim, a principal contribuição deste artigo é o detalhamento da fase de seleção de equipes tecnicamente qualificadas.

O restante do artigo está estruturado da seguinte maneira. Seção 2 detalha a abordagem proposta, levando em consideração requisitos técnicos dos módulos de software e habilidades técnicas das equipes candidatas. A Seção 3 discute trabalhos relacionados, evidenciando as contribuições da abordagem proposta. Por fim, a Seção 4 apresenta as considerações finais e trabalhos futuros, sinalizando resultados preliminares obtidos nos experimentos iniciais.

2. Uma Abordagem para Seleção de Equipes Tecnicamente Qualificadas

A abordagem proposta é dividida em quatro passos (Figura 1), que consistem em descrever os requisitos técnicos dos módulos de software e os atributos técnicos das equipes distribuídas, e, então, considerando uma política de seleção, avaliar a adequabilidade técnica de todas as equipes candidatas em relação aos módulos de software da LPS, resultando no mapeamento de conjuntos de equipes tecnicamente qualificadas para desenvolver cada módulo de software.

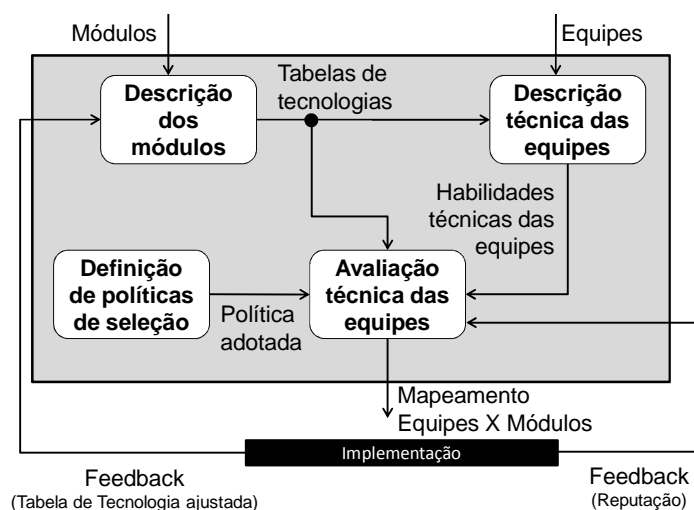


Figura 1. Visão geral da seleção técnica de equipes

2.1. Descrição dos Módulos

Na abordagem proposta, o passo denominado *Descrição dos Módulos* identifica as tecnologias requeridas para implementar cada módulo de software, bem como os seus respectivos domínios de conhecimento. Neste trabalho, um módulo de software é um conjunto de componentes com forte acoplamento intra-módulo e fraco acoplamento inter-módulo. Cada módulo constitui uma unidade de implementação, podendo ser desenvolvida de forma independente por uma equipe tecnicamente qualificada.

Como o objetivo principal é avaliar a adequabilidade técnica das equipes, para cada tecnologia e domínio de conhecimento identificado, também é necessário mensurar o nível de conhecimento requerido pelo módulo. Para tal, o artefato denominado *Tabela de Tecnologias* deve ser produzido para cada módulo, conforme ilustrado na Tabela 1. É importante mencionar que o número de níveis de conhecimento e os termos adotados para mensurá-los podem ser definidos pelo gerente de projeto, caso julgar necessário.

Tabela 1. Tabela de Tecnologias

Baixo	Médio	Alto
Eclipse	Java	XP
Windows	Domínio Médico	SQL

Na implementação dos módulos, é possível que algumas tecnologias previstas não tenham sido utilizadas na prática, ou que tecnologias utilizadas de fato não tenham sido previstas. Para lidar com tais casos, após a implementação dos módulos, informações de *feedback* providas pelo gerente são usadas para revisar e atualizar as tabelas de tecnologias, que devem ser mantidas em uma base de dados, permitindo o reuso das mesmas em futuras iterações da LPS.

Considerando que abordagens de LPS são inerentemente iterativas, podemos perceber que módulos de software previamente desenvolvidos podem ser redefinidos em futuras iterações. Nestes casos, para ser considerada tecnicamente qualificada, uma equipe deve conhecer tanto as tecnologias necessárias para implementar os novos requisitos do módulo quanto as tecnologias utilizadas na construção do módulo reusado.

2.2. Descrição Técnica das Equipes

Na abordagem proposta, o passo denominado *Descrição Técnica das Equipes* identifica o nível de conhecimento das equipes em todas as tecnologias e domínios de conhecimento requeridos para implementar os módulos da LPS. Para coletar tais informações das equipes, o gerente de projeto deve montar um questionário a partir das tabelas de tecnologias, conforme ilustrado na Tabela 2. Os valores utilizados para representar as respostas podem ser definidos pelo gerente de projeto, caso julgar necessário. Observe que, para cada tecnologia e domínio de conhecimento, três tipos de informações são coletadas: *anos de experiência*, *número de projetos desenvolvidos* e *número de títulos*. Ao processar os questionários, o gerente de projeto deve definir um peso para cada título identificado.

Tabela 2. Questionário

Tecnologia: JAVA
Anos de experiência: () Nenhum () 1-3 () 3-6 () 6+
Participação em projetos: () Nenhum () 1-5 () 5-10 () 10+
Certificações: () SCJP () SCWCD () SCJCD

De acordo com [Shanteau *et al* 2002], o número de anos de experiência, bem como o número de títulos e certificados, são comumente utilizados para avaliar se um indivíduo é especialista em um dado domínio. Complementando, em [Weiss *et al* 2006], a capacidade de discernimento, que representa a capacidade do indivíduo perceber diferenças sutis em contextos similares, é indicada como um importante fator que determina se um indivíduo é especialista em um dado domínio. Dada sua natureza empírica, a capacidade de discernimento pode ser inferida observando eventos passados. No caso de projetos de software, adotamos o número de projetos que o indivíduo participou como forma de mensurar tal capacidade.

É importante destacar que os formulários devem ser aplicados a cada membro de equipe. Uma vez que a habilidade técnica de cada membro é identificada, a habilidade técnica da equipe e na tecnologia ou domínio t , denotada por $h_{te_{e,t}}$, pode ser facilmente pela Equação (1), onde $h_{tm_{i,t}}$ é a habilidade técnica do membro i na tecnologia ou domínio t , n é a quantidade de membros da equipe, e max é a quantidade de membros da maior equipe.

$$h_{te_{e,t}} = \log_{max} n * \frac{\sum_{i=0}^n h_{tm_{i,t}}}{n} \quad (1)$$

Note que o lado direito da Equação (1) representa a média da habilidade técnica dos membros da equipe. No entanto, deve-se considerar que o conhecimento dos indivíduos pode ser complementar, e, portanto, indivíduos agregam valor ao nível de conhecimento. Logo, supõe-se que quanto mais membros existem em uma equipe mais conhecimento é compartilhado, aumentando assim o nível de conhecimento da equipe. Mas, considerando o caráter aparentemente finito do conhecimento em cada tecnologia ou domínio de conhecimento, supõe-se também que a contribuição de cada indivíduo no nível de conhecimento da equipe não é linear, tendendo a ser inexpressivo a medida que o número de indivíduos torna-se grande. Para representar o efeito complementar e não linear do conhecimento dos indivíduos, um fator de amortização é aplicado às equipes menores, representado pela função log na Equação (1).

Embora a habilidade técnica das equipes seja calculada como um número real, a subjetividade envolvida introduz certo grau de imprecisão. Para lidar com essa imprecisão, a habilidade técnica de cada equipe em cada tecnologia ou domínio é transformada e representada em conjuntos de termos vagos, como *nenhum*, *médio*, *baixo* e *alto*, conforme ilustrado na Figura 2. Esses termos vagos são na verdade conjuntos difusos que escondem a complexidade numérica do usuário. A representação da habilidade técnica das equipes com termos (conjuntos) *fuzzy* é realizada através do processo de *fuzificação*, que consiste em encontrar o nível de pertinência de conhecimento de uma equipe em cada conjunto. Desta forma, a imprecisão pode ser tratada sem perdas significativas e o valor torna-se mais compreensível ao ser humano.

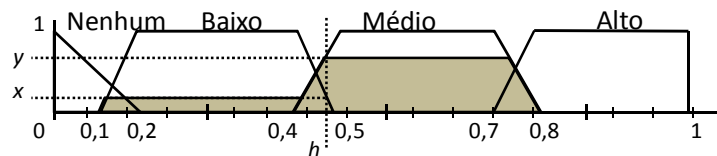


Figura 2. Conjunto *fuzzy* de conhecimento de equipes

O tamanho dos trapézios e o número de termos podem ser alterados de acordo com necessidades de cada projeto. Como exemplo, suponha que uma equipe possua habilidade técnica h em uma determinada tecnologia. Na Figura 2, h corta dois conjuntos: *baixo* (valor de pertinência x) e *médio* (valor de pertinência y). Como $y > x$, é possível dizer que a habilidade técnica da equipe é mais média que baixa ou que está mais próximo de média, embora os dois termos sejam utilizados simultaneamente para representar a habilidade técnica.

2.3. Definição de Políticas de Seleção

Antes de iniciar o processo de avaliação das equipes, uma política de seleção deve ser definida usando uma *tabela de regras*. A Tabela 3 apresenta um exemplo onde: colunas representam os níveis de conhecimento requeridos para uma dada tecnologia ou domínio; linhas representam os níveis de habilidade técnica da equipe naquela tecnologia ou domínio; e células representam o *nível de adequabilidade* a ser inferido para a equipe, que também é definido usando conjuntos difusos. A tabela de regras pode ser interpretada como um conjunto de regras IF-THEN. Por exemplo: *IF nível de conhecimento IS baixo AND habilidade técnica da equipe IS baixa THEN nível de adequabilidade IS alto*. Lembre-se que a habilidade técnica da equipe pode ser representada por mais de um termo simultaneamente, como mostrado na Figura 2. Neste caso, mais de uma regra pode ser disparada.

Tabela 3. Política de seleção

		Nível de conhecimento na tecnologia ou domínio		
		Baixo	Médio	Alto
Habilidade técnica da equipe	Nenhum	Nenhum	Nenhum	Nenhum
	Baixo	Alto	Médio	Baixo
	Médio	Médio	Alto	Médio
	Alto	Baixo	Médio	Alto

Dependendo das necessidades do projeto, diferentes regras podem ser definidas, resultando em critérios de seleção diferentes. Por exemplo, como mostrado na Tabela 3,

pode-se definir uma política que seleciona as equipes cujas habilidades técnicas são mais próximas do nível de conhecimento requerido para implementar os módulos. Esse tipo de política poupa equipes mais qualificadas para módulos mais complexos. No entanto, se o projeto está atrasado, pode ser conveniente selecionar as equipes com as qualificações mais altas, não importando a complexidade dos módulos. Embora esta decisão possa acelerar o projeto, é provável que aumente seu custo.

2.4. Avaliação Técnica das Equipes

Considerando a descrição técnica dos módulos (nível de conhecimento requerido por cada módulo em cada tecnologia e domínio), a descrição técnica das equipes (nível de habilidade técnica de cada equipe em cada tecnologia ou domínio) e a política de seleção adotada, no passo denominado *Avaliação Técnica das Equipes* é possível calcular a adequabilidade técnica da equipe para cada tecnologia ou domínio.

Como várias tecnologias ou domínios podem ser requeridas na implementação do módulo, vários níveis de adequabilidade são gerados. Contudo, como o objetivo final é apresentar uma métrica única de adequabilidade, é necessário transformar cada termo difuso em uma representação numérica. Isto é feito através da *defuzzificação* dos termos utilizando-se o método *Centro das Somas* [Ross 2010], que gera um valor único no intervalo [0, 1].

Em seguida, a adequabilidade da equipe para implementar um módulo é calculada como sendo a média dos valores numéricos da adequabilidade em cada tecnologia ou domínio requerido pelo módulo. Este processo é repetido até que se descubra a adequabilidade de cada equipe para cada módulo. Uma vez que a adequabilidade das equipes aos módulos é calculada, é possível classificar as equipes em ordem decrescente de adequabilidade para cada módulo.

Considerando que a abordagem proposta trata-se de um processo de seleção, o gerente de projeto deve definir um *ponto de corte*, cujo efeito é eliminar as equipes cujas adequabilidades estão abaixo desse ponto de corte. Considerando que a adequabilidade da equipe é calculada para cada módulo, uma dada equipe pode não entrar no conjunto selecionado para determinados módulos, mas ser mantida no conjunto selecionado para outros módulos.

Neste ponto, assegurado que as equipes restantes possuem o nível mínimo de adequabilidade técnica necessária para implementar os módulos, o atributo denominado *reputação* agora é considerado para reordenar as equipes. A reputação de uma equipe é um indicador histórico da qualidade técnica das equipes na prática, representando quão bem a equipe realizou o desenvolvimento de outros módulos em iterações passadas do atual projeto ou mesmo em outros projetos. Já que o processo de descoberta de conhecimento é inerentemente subjetivo, a reputação indiretamente trata aspectos de desempenho, consumo de tempo e qualidade técnica das equipes na prática.

Confirme ilustrado na Figura 1, após a implementação dos módulos, o gerente de projetos avalia o desempenho de cada equipe, podendo bonificá-las, penalizá-las ou manter-se neutro, e, assim, aumentar ou diminuir a reputação das mesmas. A reputação assume valores no intervalo [0, 1], e, inicialmente, todas as equipes possuem reputação neutra, representado pelo valor 0,5. Valores acima de 0,5 representam boa reputação. Inversamente, valores abaixo de 0,5 representam má reputação.

Considerando a adequabilidade e a reputação da equipe, o ajuste final da adequabilidade da equipe é realizado usando a Equação (2), onde $A_{e,m}$ representa a adequabilidade técnica da equipe e ao módulo m , e R_e representa a reputação da equipe e . Como resultado, por um lado, a adequabilidade é mantida inalterada para reputação neutra com valor igual a 0,5. Por outro lado, a adequabilidade é levemente melhorada ou piorada para de reputação boa ($> 0,5$) ou má ($< 0,5$), respectivamente.

$$A_{e,m} = A_{e,m}^{R_e} \quad (2)$$

Por fim, após realizar o ajuste da adequabilidade, as equipes são reordenadas e o resultado final representa as equipes tecnicamente qualificadas que foram selecionadas. Na Figura 1, o artefato *Mapeamento Equipes X Módulos* representa o conjunto de equipes tecnicamente qualificadas a implementar cada módulo. Note que, de posse desse mapeamento, o gerente de projeto pode utilizar sua experiência e conhecimento para alocar as equipes de desenvolvimento com muito mais propriedade e confiança.

3. Trabalhos Relacionados

Como primeiro trabalho relacionado, [Barreto *et al* 2005] apresenta uma abordagem que utiliza a técnica de satisfação de restrição para selecionar profissionais para tarefas de projetos de software. Funções utilitárias são definidas, as quais devem ser maximizadas ou minimizadas pelos profissionais selecionados, gerando assim a solução de seleção. As restrições servem como um filtro na seleção já que apenas os profissionais que as satisfazem são considerados no processo de seleção.

Em [Duggan *et al* 2004], é apresentada uma abordagem que utiliza algoritmos genéticos para selecionar profissionais para as atividades de implementação de um projeto de software. Soluções são geradas de maneira a otimizar aspectos específicos, tais como custo de desenvolvimento, *time-to-market* ou diminuição de defeitos. Assim, embora permita encontrar diversas soluções, cada uma apresenta uma otimização de um determinado aspecto.

Uma abordagem baseada em lógica *fuzzy* é apresentada em [Callegari *et al* 2009]. Nesta abordagem, de forma similar a abordagem aqui proposta, o processo de seleção usa a habilidade técnica dos profissionais, o nível de dificuldade esperado para cada tarefa a ser executada, e um conjunto de regras configuráveis que é utilizado para orientar o processo de seleção. [Shen *et al* 2003] propõe outra abordagem baseada em lógica *fuzzy* que considera o relacionamento social entre os profissionais. No entanto, as regras e termos *fuzzy* usados para gerar as soluções não são configuráveis pelo gerente do projeto, tornando-a restrita a uma única política de seleção.

A abordagem aqui proposta difere das demais propostas, pois considera uma retro-alimentação das informações de reputação das equipes e atualização das tabelas de tecnologias que podem ser reusadas em novas iterações do processo de desenvolvimento. Além disso, deve-se notar que nenhum dos trabalhos apresentados fornece uma maneira sistemática de identificar e mensurar as habilidades técnicas dos membros das equipes. Nos trabalhos citados, o gerente de projetos é responsável por definir unilateral e intuitivamente o nível de conhecimentos dos profissionais. Nenhum trabalho apresentado considera também a seleção de equipes, apenas profissionais isolados para assim montar uma equipe única, tornando-se assim seu uso um pouco restrito num contexto de projetos de DDS.

4. Considerações Finais

Conforme discutido, existem diversas abordagens para selecionar profissionais para tarefas de projeto de software. No entanto, a maioria delas não provê mecanismos satisfatórios para representar a imprecisão e a subjetividade, inerentes à representação do conhecimento necessário para modelar problemas de seleção de profissionais. Assim, na abordagem proposta neste artigo, a adoção da lógica *fuzzy* torna possível ajudar os gerentes de projeto no processo de seleção, já que os mesmos passam a lidar com termos vagos utilizados no seu dia-a-dia, escondendo toda a complexidade numérica.

Para avaliar e refinar a abordagem propostas, experimentos iniciais estão sendo realizados com duas LPS: (i) um ambiente de comércio eletrônico; e (ii) um *middleware* para computação móvel. Para simular as equipes, foram elaborados questionários para coletar informações de habilidade técnica de 56 participantes. Equipes simuladas foram formadas a partir da seleção aleatória destes participantes. Além disso, 4 políticas de seleção distintas foram configuradas. Resultados preliminares evidenciam que a abordagem proposta realiza o processo de seleção de forma adequada, pois as seleções recomendadas são satisfatórias do ponto de vista das políticas de seleção adotadas. Contudo, o cálculo de reputação deve ser refinado, pois o mesmo parece influenciar de forma demasiada no cálculo da adequabilidade, aproximando excessivamente os valores de adequabilidades de algumas equipes. Após a validação e refinamento da abordagem, é planejado como trabalho futuro o desenvolvimento de ferramental para automatizá-la.

Referências

- Barreto A., Barros M., e Werner C. (2005) “Staffing a Software Project: a Constraint Satisfaction Approach”, Proceedings of the 7th International Workshop on Economics-driven Software Engineering Research, ACM, New York, pp 1-5.
- Callegari D. A. e Bastos R. M. (2009) “A Multi-criteria Resource Selection Method for Software Projects Using Fuzzy Logic”, Proceeding of the 11th International Conference on Enterprise Information Systems, Springer, Italy, pp 376-388.
- Clements P., e Northrop L. (2002). Software Product Lines: Practices and Patterns, Addison-Wesley, Boston.
- Duggan J., Byrne J., e Lyons G. (2004) “A Task Allocation Optimizer for Software Construction”, IEEE Software, California, 21 (3), pp 76-82.
- Karolak D. W. (1999) Global Software Development - Managing Virtual Teams and Environments, IEEE Computer Society, California.
- Ross T. J. (2010) Fuzzy Logic - With Engineering Applications, 3rd ed, Wiley.
- Santos, V., et al (2010) “Um Framework de Recomendação para Alocação de Equipes de Desenvolvimento em Projetos Distribuídos de Linhas de Produto de Software”, IV Workshop de Desenvolvimento Distribuído de Software (WDDS), Brasil.
- Schwalbe K. (2002) Information Technology Project Management, Thomson Learning.
- Shanteau J., et al (2002) “Performance-based assessment of expertise: How to decide if someone is an expert or not”, European Journal of Operational Research, Elsevier, 136, pp 253-263.
- Shen M., Tzeng G.-H, e Liu D.-R. (2003) “Multi-Criteria Task Assignment in Workflow Management Systems” 36th Annual Hawaii International Conference on System Sciences, pp 1-9.
- Weiss D. J., Shanteau J., e Harries P. (2006) “People who judge people”, Journal of Behavioral Decision Making, 19, pp. 441-454.
- Zadeh L. A. (1965) “Fuzzy Sets”, Information and Control, 8(3), pp 338-353.