

DESENVOLVIMENTO DE SISTEMAS AEROESPACIAIS COM O USO DE MODELAGEM E SIMULAÇÃO COMPUTACIONAL

AEROSPACE SYSTEMS DEVELOPMENT USING MODELING AND COMPUTER SIMULATION

Antonio Cesar Galhardi¹

Carlos Eduardo Cassiano Pinto²

RESUMO

O artigo discute os conceitos de *Model-Based-Design* – *MBD* e *Model Based Testing* – *MBT*, e como estes podem auxiliar na diminuição do ciclo de vida do desenvolvimento do produto, com consequente redução de custos, e otimizar o "time-to-market". Os processos tradicionais de desenvolvimento de sistemas são geralmente focados em documentações formais e estáticas. Por outro lado, as empresas mais inovadoras introduzem novas técnicas e metodologias para o desenvolvimento de sistemas, afim de melhor entenderem o problema à ser solucionado. O artigo teve como metodologia uma aplicação prática para se determinar o melhor tempo de acomodação do controlador principal do sistema de ar condicionado de uma aeronave executiva de pequeno porte da EMBRAER. Além do resultado obtido na empresa, o referencial teórico utilizado permitiu verificar que as empresas que implementaram técnicas inovadoras como: "*Model-Based-Design*" e "*Model-Based-Testing*" obtiveram ganhos significativos em todas as etapas do processo de desenvolvimento do produto.

Palavras-chave: Projetos baseados em modelos, Testes baseados em modelos, Modelagem, Simulação.

ABSTRACT

This article discusses the concepts of *Model-Based-Design* - *MBD* and *Model Based Testing* - *MBT*, and how they can assist in shortening the product development lifecycle, thereby reducing costs, and optimizing time-to- market ". Traditional systems development processes are usually focused on formal and static documentation. On the other hand, the most innovative companies introduce new techniques and methodologies for systems development, in order to better understand the problem to be solved. The article had as methodology a practical application to determine the best time of accommodation of the main controller of the air conditioning system of an EMBRAER small business aircraft. In addition to the result obtained in the company, the Theoretical Framework used showed that companies that implemented innovative techniques such as: "*Model-Based-Design*" and "*Model-Based-Testing*" achieved significant gains in all stages of the product development process.

Keywords: Model-based-design, Model-based-testing, Modeling, Simulation.

¹ Doutor em engenharia mecânica Universidade Federal de São Carlos. Professor em programa de mestrado profissional do Centro Paula Souza. E-mail: galhardi.antoniocesar@gmail.com.

² Mestrando do programa de mestrado profissional do Centro Paula Souza. E-mail: du.cassiano@uol.com.br.

1 INTRODUÇÃO

A utilização de técnicas de testes de sistemas tem permitido melhorar a eficiência no desenvolvimento do produto, tão fundamental para a sobrevivência das empresas altamente competitivas e que desenvolvem sistemas complexos; isto porque elas necessitam uma vez que de métodos sistemáticos de aplicação de testes durante o projeto.

No entanto, ainda são poucas as aplicações práticas na elaboração de testes automáticos, principalmente no Brasil. Apesar da literatura internacional apontar o quanto os testes automáticos agregam valor ao desenvolvimento de produto, e como eles podem ser realizados de forma organizada e eficiente durante todo o processo de desenvolvimento do produto.

O desenvolvimento de sistemas complexos como, as aeronaves, requerem métodos sistemáticos para o seu desenvolvimento e projeto. Tipicamente, o desenvolvimento se inicia com a captura de requisitos de alto nível, que são registrados em formato textual. Estes requisitos geralmente formam a base para a especificação dos sistemas; que posteriormente são gradualmente refinados e implementados, a partir de um projeto detalhado.

Durante a implementação do projeto detalhado, os engenheiros projetistas utilizam cenários de testes estruturados para validarem os sistemas desenvolvidos e todo o seu ambiente contra os requisitos originais previamente definidos. A construção de sistemas complexos requer alta exigência em segurança, o que por sua vez exige rigorosos processos de especificação do comportamento do sistema. Alia-se a isto, uma campanha de testes para garantir que a implementação satisfaça totalmente os requisitos do sistema.

A partir deste cenário, as técnicas de modelagem e simulação tornaram-se um fator significativo do Processo de Desenvolvimento do Produto. Tais técnicas são conhecidas como: *Model-Based-Design – MBD* e *Model-Based-Testing – MBT*, que podem ser aplicadas concomitantemente para reduzir o custo do projeto e até mesmo o tempo do *Time-to-Marketing*, principalmente quando aplicadas desde o início do Processo de Desenvolvimento do Produto.

Para a maior facilidade de se exercitar o sistema em desenvolvimento por meio de modelos, é comum a “explosão” no número de testes possíveis e desejáveis a serem realizados durante as várias etapas do desenvolvimento, desde a validação de requisitos até os testes finais. Isto permite a elaboração de “*frameworks*”; além de recursos computacionais de hardware e software para suportar tais aplicações.

O objetivo geral do artigo é caracterizar os processos de desenvolvimento de sistemas complexos, particularmente o *Model-Based-Design – MBD* e o *Model-Based-Testing – MBT*, em uma aeronave comercial de médio porte, da EMBRAER.

As perguntas relacionadas à investigação se estabelecem como:

- a) Os conceitos de “*Model-Based-Design*” (MBD) e “*Model-Based-Testing*” (MBT) são efetivos em quais etapas do processo de desenvolvimento do sistema?
- b) Como planejar estes testes ao longo do desenvolvimento do produto?

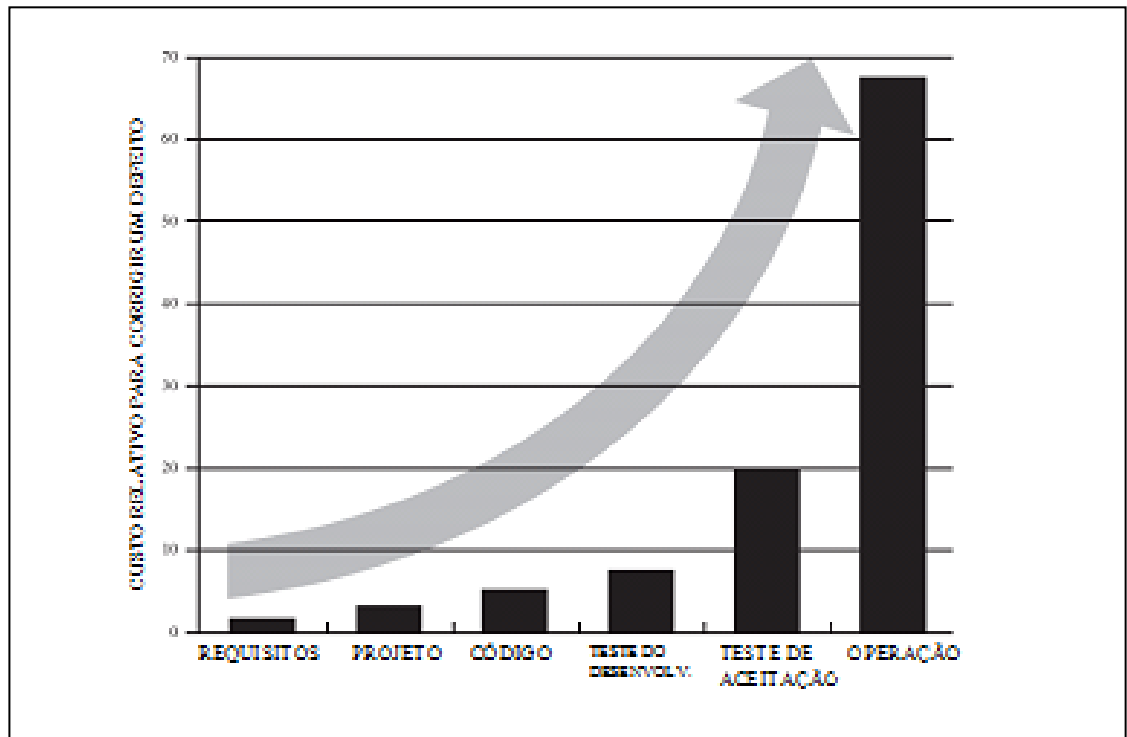
2 REFERENCIAL TEÓRICO

Com o avanço no desenvolvimento de aplicações de sistemas embarcados as técnicas de MBD e MBT, normalmente incorporadas no desenvolvimento deste tipo de produtos, se tornaram cada vez mais difundidas. O desenvolvimento de aplicações embarcadas normalmente exige: automação dos testes devido à grande quantidade de testes requerida; projetos e testes orientados a modelos; modelagem matemática para atender aos requisitos de confiabilidade e desempenho; e componentes desenvolvidos para reuso (GRECA, 1999).

Os processos baseados em modelagem e simulação têm sido largamente utilizados no desenvolvimento de sistemas complexos, a fim de minimizar a quantidade de testes a serem realizados em protótipos, e que usualmente possuem custo elevado, além de agilizar as fases de desenvolvimento do processo. Atualmente a EMBRAER, fabricante brasileira de aeronaves de pequeno e médio porte utiliza tal tecnologia em diversas etapas do desenvolvimento do produto, desde modelos para ensaio em túnel de vento, como modelos para prototipação rápida de sistemas, e em ferramentas de simulação *Computer Aided Engineering – CAE* (SILVA 2003). Eles são a contrapartida aos desenvolvimentos tradicionais, cuja detecção de defeitos se dá por todo o período do desenvolvimento (BURNSTEIN, 2003).

Isto pode ser visto na Figura 1. Uma figura clássica na literatura que demonstra a evolução do custo da detecção de um erro ao longo do desenvolvimento do produto de software, utilizando como base um processo tradicional de desenvolvimento de sistemas.

Figura 1 - Custo de detecção de defeitos ao longo do desenvolvimento



Fonte: adaptado de Burnstein (2003) - *Practical Software Testing*

Pode-se dizer que tudo teve início com o Projeto Condor: criado em 1985 é um exemplo típico de uma de *Distributed High Throughput Computing*, e especializado no *High-Throughput Computing* clássico. O termo de Ciência da Computação é utilizado para descrever o uso de muitos recursos computacionais durante longos períodos de tempo, e para realização de uma tarefa computacional (THAIN, TANNENBAUM e LIVNY, 2005).

O projeto descrito por computacional Thain, Tannenbaum e Livny, (2005) consiste em um time de aproximadamente 35 professores, funcionários em tempo integral e estudantes que: enfrentam os desafios de engenharia de software em um ambiente UNIX / Linux / NT distribuído, estão envolvidos em colaborações em grids nacionais e internacionais, interagem ativamente com entidades e usuários acadêmicos e comerciais, mantem e suportam grandes ambientes distribuídos em produção e educam e treinam estudantes. É um sistema integrante de muitos grids computacionais ao redor do mundo. O sistema condor é um sistema repleto de “*features*” como: mecanismo de políticas de agendamento flexíveis via ClassAds, preempção, suspensão, requisitos, preferências, grupos, quotas, *fair-share* (compartilhamento), *system*

hold, facilidades para gerenciar tanto CPUs dedicadas (clusters) quanto recursos não dedicados (desktops), não são necessário um “*shared file-system*”, Gerenciamento de Workflow (interdependências de tarefas), Suporte a muitos tipos de *jobs* – serial e paralelo Tolerante a falha: pode sobreviver a falhas, interrupções de rede, sem ponto único de falha, APIs de desenvolvimento: via SOAP / *web services*, DRMAA (C), Perl *package*, GAHP, *flexible command-line tools*, MW (*Master Worker*) para plataformas Linux, Windows, MacOS, Solaris, HP-UX, AI.

O contexto da computação distribuída tratado neste artigo utiliza o conceito de *Distributed High Throughput Computing* – DHTC, que trata de sua efetiva utilização para o gerenciamento e a exploração de todo recurso de computação disponível.

Segundo o Escritório de Modelagem e Simulação do Departamento de Defesa dos Estados Unidos: os modelos são representações físicas, matemáticas ou lógicas de um sistema, entidade, fenômeno ou processo. Modelar consiste basicamente em aplicar padrões e rigorosas metodologias para a criação e validação de um modelo. Esta afirmação refere-se aos modelos de forma geral; no entanto, no que se refere a modelagem e simulação em engenharia, é possível se destacar algumas aplicações como: desenvolvimento de controladores, especificação e validação de requisitos de sistemas, simulação computacional, cálculos estruturais, entre outras (DEFENSE, 2007).

Para Althoff (2007) o uso das técnicas de “*Model-Based-Design*” e “*Model-Based-Testing*” vem sendo altamente adotadas no desenvolvimento de sistemas computacionais embarcados. Diversas companhias têm observado as vantagens obtidas nos processos de desenvolvimento como: diminuição do tempo de desenvolvimento, aumento da qualidade final do produto e significativa redução de custos. O mesmo autor ressalta que um dos grandes benefícios da utilização de modelos é a capacidade de desenvolver algo que pode ser entendido e utilizado por todas as pessoas envolvidas num projeto, facilitando o reuso e testes; e que podem ser tratados como especificações executáveis do sistema.

Os processos de desenvolvimento baseados em modelos e testes, MBD e MBT, possibilitam desenvolver diversas soluções alternativas para validar a melhor tomada de decisão de projeto, utilizando-se de laboratórios e simulações. (BAKER e LONG, 2006).

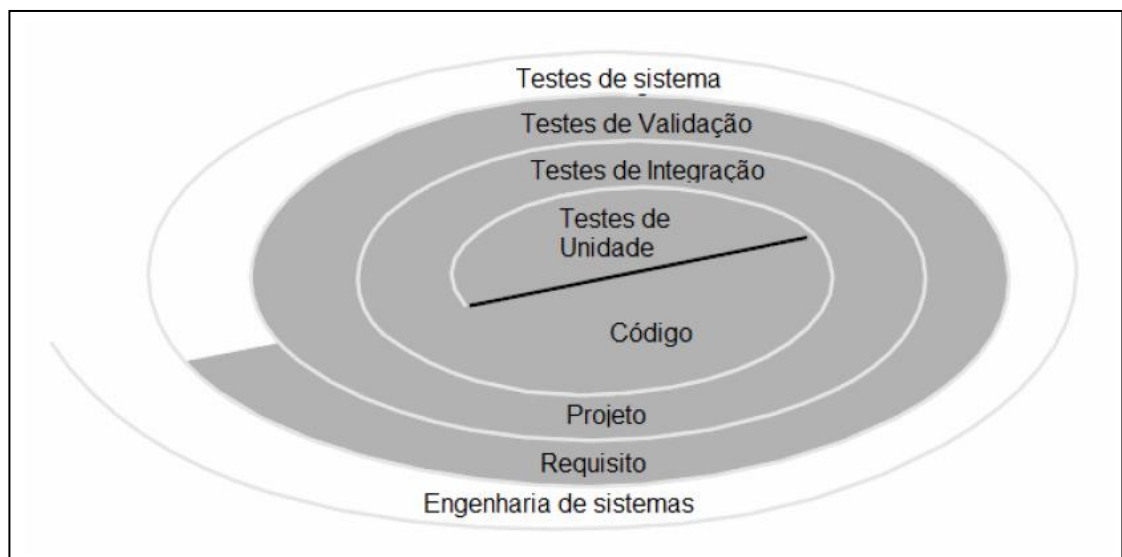
O planejamento do processo de validação e verificação normalmente é realizado por meio de um Plano de Testes, que é um documento que apresenta: uma breve introdução específica, os tipos de testes que serão executados ao longo do processo de desenvolvimento

do produto, os objetivos destes testes, a metodologia utilizada, as técnicas e tipos de teste a serem realizados (BURNSTEIN, 2009).

O Plano de Teste dá suporte para atingir os objetivos dos testes, e geralmente é desenvolvido por um especialista de testes ou de qualidade (VIJAYKUMAR, 2009).

Para Pressman (2007) verificar com provas objetivas que os requisitos especificados foram cumpridos; é validar o processo. O autor ressalta que o processo de validação é muito utilizado durante a fase de levantamento de requisitos, para certificar que o sistema a ser construído irá ser capaz de atender aos requisitos de mercado. Já na verificação ela ocorre praticamente em todas as fases do desenvolvimento, para garantir que os produtos de uma dada fase implementem em sua totalidade as entradas e saídas especificadas; ou seja: o produto foi construído corretamente. O autor apresenta ainda a estratégia em espiral para desenvolvimento de software, como apresentado na Figura 2.

Figura 2 - Estratégia de teste de software de forma espiral



Fonte: adaptado de Pressman (2007) - A Practitioner's approach

Para Ferraz (2010), os modelos podem ser entendidos como uma simulação computacional do comportamento de algum sistema real ou imaginário, ao longo do tempo. Tais simulações são amplamente utilizadas para análise de sistemas e processos dinâmicos, como controle de tráfego aéreo, redes de comunicação, processos físicos e biológicos, e no desenvolvimento do projeto de sistemas em geral.

No final da década de 1970, 50% do tempo e 50% do custo de um projeto eram destinados para testar programas ou sistemas. Hoje, apesar de linguagens de programação mais avançadas, e novos ambientes de desenvolvimento; os testes continuam tendo um papel fundamental para garantir a qualidade do produto. Para Myers (2010), um bom processo de validação e verificação é atingido por meio de testes. A citação do mesmo autor: teste e depois codifique, mostra claramente a importância da realização de testes em todas as etapas do desenvolvimento do software; um conceito que fundamenta a presente pesquisa, no sentido de que a atividade de testes não seja custosa.

3 MÉTODO

Utilizou-se uma aplicação prática para se determinar o melhor tempo de acomodação do controlador principal do sistema de ar condicionado de uma aeronave executiva de pequeno porte, associado ao relato de aplicação do *grid-computing*, na fabricação de uma aeronave comercial de médio porte, da EMBRAER. Esta pesquisa caracteriza-se por uma metodologia Qualitativa-Exploratória-Descritiva, associado ao relato de aplicação do *grid-computing*.

O Referencial Teórico adotado contempla os conceitos e definições relacionados ao tema, além de exemplos de aplicações do *Model-Based-Design*, em do ramo aeronáutico e de sistemas embarcados.

A análise das aplicações nas etapas do processo de desenvolvimento do produto e os resultados obtidos permitiram obter subsídios para encontrar as respostas as perguntas básicas de pesquisa.

4 RESULTADOS E DISCUSSÃO

Os principais objetivos da utilização de MBD e MBT são: redução do ciclo de desenvolvimento do produto; a antecipação da detecção de defeitos; facilitar a compreensão dos testes de sistemas; reuso dos artefatos de teste ao longo das etapas de desenvolvimento.

Os modelos geralmente seguem os mesmos níveis de abstração da etapa de desenvolvimento, por exemplo, durante a etapa de elaboração e validação de requisitos não se espera, já nesta fase, utilizar modelos detalhados do sistema e sim modelos que representem o mesmo nível de abstração dos requisitos.

A utilização de MBD e MBT possibilita a escrita de testes efetivos para todas as principais etapas de desenvolvimento do produto; desde o início do desenvolvimento na especificação de requisitos por meio de modelos, já é possível identificar a dificuldade de implementação de determinados requisitos.

Desenvolver testes para validação e verificação de sistemas complexos exige um grande conhecimento sobre o sistema desenvolvido. Desenvolvendo os testes em paralelo com o projeto e desenvolvimento do sistema, torna possível o maior engajamento do engenheiro de testes, ainda nas fases preliminares do desenvolvimento do sistema, facilitando o processo de Validação e Verificação. Isto pois, tal colaboração resulta numa verificação antecipada do produto reduzindo custos e aumentando o tempo para correção de erros sem afetar a entrega final do produto.

O reuso de testes é uma palavra chave em se tratando da metodologia "*Model-Based-Design*". A capacidade de executar os mesmos testes já realizados por meio de modelos nos hardwares reais, possibilita saber exatamente como os componentes físicos devem se comportar no laboratório. E se os testes encontrarem valores inesperados, o reuso dos testes possibilita de forma fácil e rápida, interagir com o projeto e solucionar o problema.

O tempo e o custo geralmente são os limitantes da quantidade e variabilidade de características testadas no produto. A possibilidade de realização de testes em ambientes de simulação possibilita a criação de cenários de teste de forma rápida, e até mesmo em paralelo, explorando-se os domínios do problema em laboratórios. O ciclo de desenvolvimento utilizando as técnicas de MBD e MBT permite a validação e verificação continua ao longo de todas as etapas do processo, conforme descrito anteriormente.

Com a execução do desenvolvimento, os modelos e a arquitetura do produto vão adquirindo um nível de fidelidade maior, o que possibilita outras etapas de validação, como testes de integração e testes de sistemas. Já nas etapas finais de desenvolvimento, as técnicas como "*Hardware-in-the-loop - HIL*" possibilitam a validação e verificação em ambientes simulados, com cenários bem próximos ao ambiente real em que o produto final irá ser submetido aos testes.

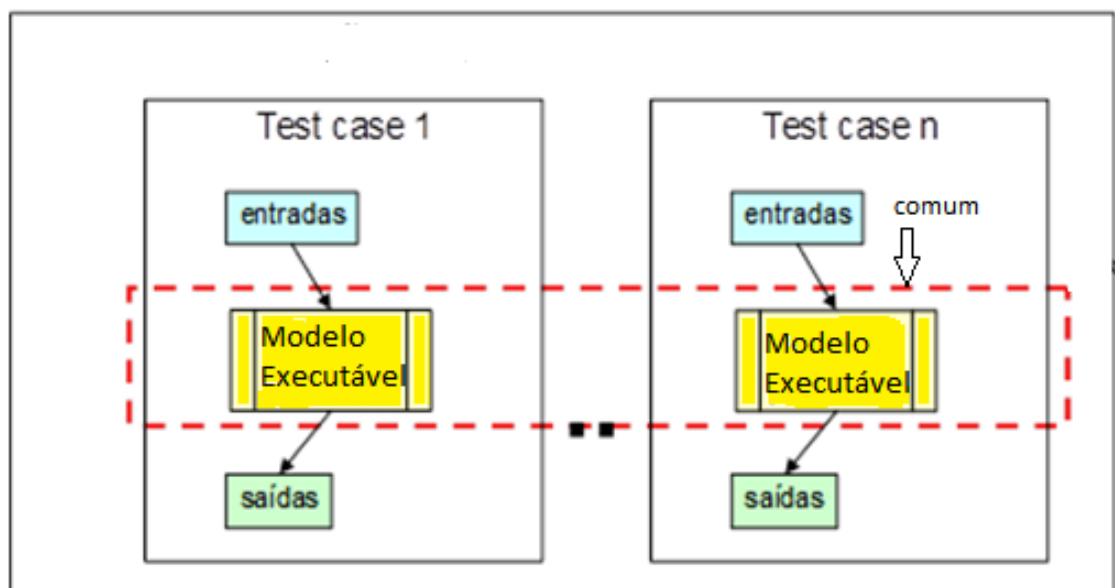
Com o crescimento do número de testes, os recursos computacionais de hardware e software passaram a ser gargalos. Assim a computação paralela, graças ao seu poder computacional, tem sido de grande auxílio na solução deste problema.

Atualmente as grandes empresas possuem um parque de *desktops* de engenharia que, em geral, ficam inoperantes, durante grande parte do dia. Esses recursos de hardware podem ser utilizados para realizar um grande número de testes. Assim estabelece-se um potencial de processamento na forma de “*grid-computing*”, que pode ser utilizado em baixo acoplamento; ou seja: onde as tarefas de execução dos casos de teste são divididas entre os membros do *grid*.

A Figura 3, apresenta os casos de testes e o modelo de simulação distribuído entre os nós da rede; ou seja: nos *desktops* de engenharia. Observa-se ainda que o modelo passa a ser sempre o mesmo em todos os casos.

Uma aplicação prática para se determinar o melhor tempo de acomodação do controlador principal do sistema de ar condicionado de uma aeronave executiva de pequeno porte, apresentava a necessidade de exercitar a varredura de cinco parâmetros do controlador, com no mínimo vinte pontos por parâmetro, o que resulta em cem casos de simulação; como cada caso demandava 10,2 minutos para sua execução, requer-se um total de 17h de simulação, sem o emprego da técnica de *grid-computing*.

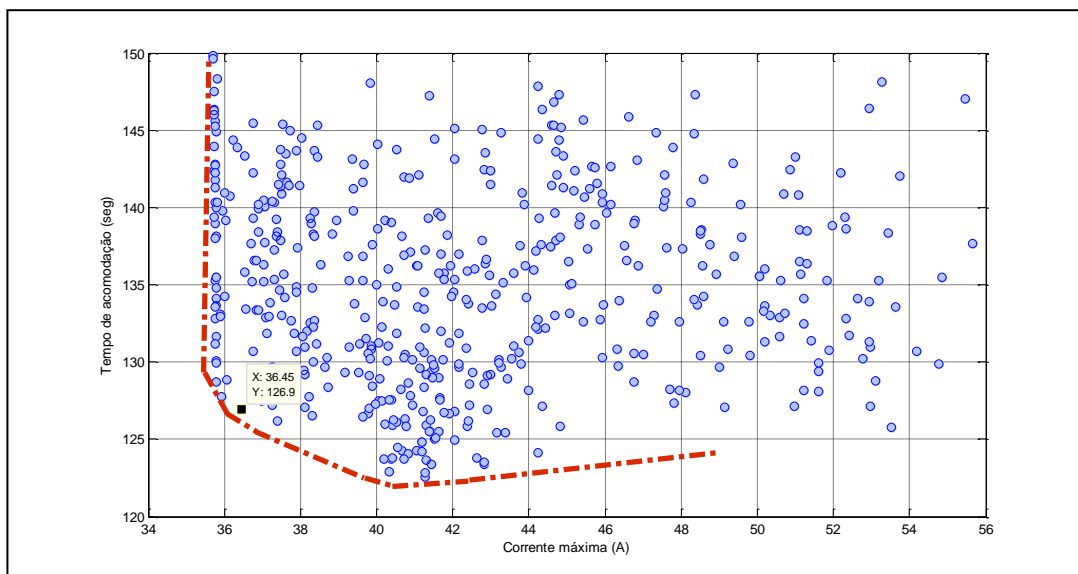
Figura 3 – Distribuição da tarefa de teste entre os nós do “*grid-computing*”



Fonte: os autores

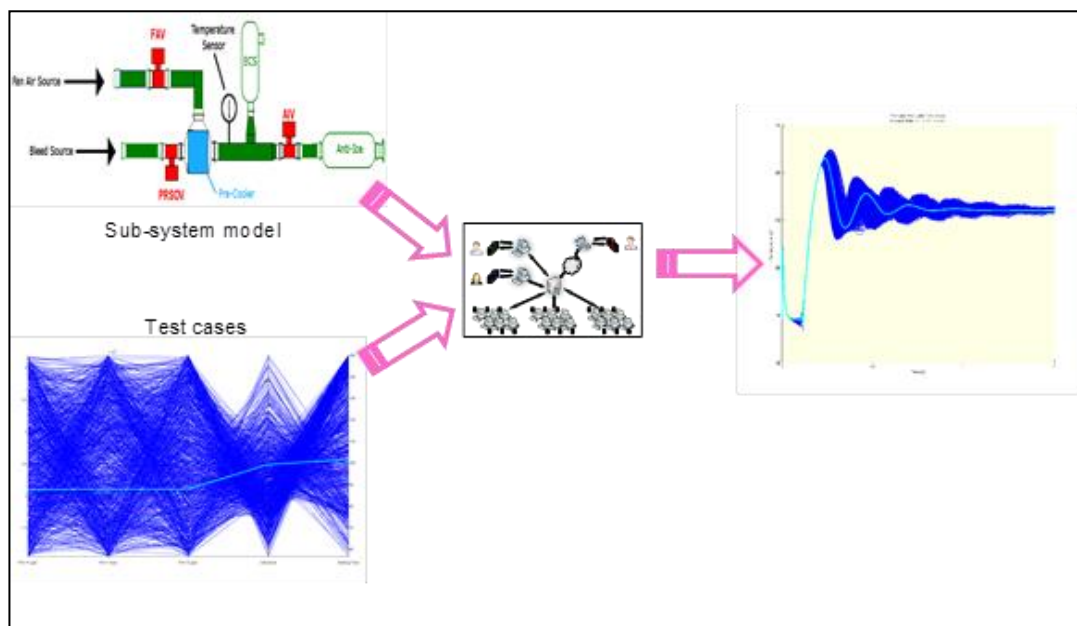
Com a utilização do *grid computing* para realizar as simulações, obteve-se uma redução de 95% do tempo gasto, em um parque de 18 desktops de engenharia. Na Figura 4 observa-se a fronteira de Pareto dos resultados das simulações de teste; enquanto a Figura 5 ilustra a arquitetura utilizada para determinar o melhor ajuste dos ganhos do controlador.

Figura 4 – Fronteira de Pareto dos resultados dos testes



Fonte: os autores

Figura 5 – Filosofia utilizada para execução dos casos de teste por meio do “*grid-computing*”



Fonte: os autores

5 CONSIDERAÇÕES FINAIS

Referencial Teórico utilizado nesta pesquisa, apresentou que as empresas que implementaram as técnicas inovadoras de “*Model-Based-Design*” e “*Model-Based-Testing*” obtiveram ganhos significativos em todas as etapas do processo de desenvolvimento do produto.

No relato de estudo da Embraer evidenciou-se que o envolvimento do engenheiro de testes desde o início do desenvolvimento, permite antecipar a detecção de defeitos e reduzindo custos, mostrou-se viável, pelas técnicas de “*Model-Based-Testing*”.

Foi possível verificar a importância do planejamento dos testes ao longo do desenvolvimento a fim de garantir a maturidade do produto desenvolvido.

Por fim, ficaram evidenciados os ganhos da utilização da técnica de “*grid-computing*” na execução dos testes em ambiente simulado.

6 REFERÊNCIAS

ALTHOLFF, G. F., Using executable assertions for runtime fault detection in a Model-Based Software Development Approach. Dissertação de Mestrado, ITA, SP, 2007.

BURNSTEIN, I. Practical Software Testing, Springer, EUA 2003.

DEFENSE, MIL-STD-882-SD, standard practice for system safety. Fevereiro 2007.

FERRAZ, R. B. D., Proposta de Ferramenta para desenvolvimento de simuladores, Dissertação de Mestrado, USP, SP, 2010.

MYERS, G. J., The Art of Software Testing: John Wiley & Sons, 2010.

PRESSMAN, R., Software Engineering: A Practitioner’s Approach, 5ª edition, EUA 2007.

SILVA, E. T., “*Quantitative Methods for management decisions in flight test campaigns for aeronautical certification*”. Dissertação de Mestrado, ITA, SP, 2003.

THAIN D., TANNENBAUM T., LIVNY M., "*Distributed Computing in Practice: The Condor Experience*" *Concurrency and Computation: Practice and Experience*, Vol. 17, No. 2-4, pages 323-356, EUA, April 2005.