

Arquitetura de Controle de Alto Nível para Barcos Robóticos

Einstein G. dos Santos¹, Rafael V. Aroca¹ e Luiz M. G. Gonçalves¹

¹Universidade Federal do Rio Grande do Norte
DCA-CT-UFRN, Campus Universitário, Lagoa Nova, 59072-970, Natal, RN, Brazil

einsantos@ymail.com, rafaelaroca@ieee.org, lmarcos@natalnet.br

Resumo—Propomos uma arquitetura de controle de alto nível para barcos robóticos destinados a atuarem em ambientes aquáticos diversos, de forma totalmente autônoma, realizando telemetria com uma estação-base e desta recebendo missões a serem realizadas. Tal proposta visa aplicar-se dentro do projeto N-Boat do laboratório Natalnet-DCA, que tem como objetivo principal capacitar um veleiro a navegar autonomamente. Os componentes constituintes dessa arquitetura são os módulos de memória, estratégia, comunicação, sensoriamento, atuação, energia, segurança e supervisão, formando estes os sistemas do barco e da estação-base. Para sua validação foi desenvolvido um simulador em *software*, na linguagem C, cujos principais resultados foram obtidos na implementação dos módulos de memória e de estratégia, mais especificamente no compartilhamento de dados e no planejamento de rotas baseado em um algoritmo de navegação, respectivamente. Esses resultados estimulam, como próximo passo, a implementação da arquitetura desenvolvida em barcos reais.

Palavras-chave: Plataformas Robóticas Autônomas, Veleiros Autônomos, Arquitetura de controle.

I. INTRODUÇÃO

Neste artigo é proposta uma arquitetura de controle de alto nível para uma plataforma aquática robótica, mas especificamente um veleiro, com capacidade de locomoção autônoma, ou seja, uma embarcação robótica equipada com sensores, atuadores e sistemas de comunicação para envio de telemetria e recepção de comandos e missões a serem executadas sem a necessidade de intervenção humana, tendo como propulsão a força dos ventos, captada por velas, e motores elétricos, alimentados por baterias carregadas por painéis solares.

Na atual fase da pesquisa a arquitetura de controle desenvolvida será utilizada em um protótipo, um nautimodelo rádio-controlado, dotada de: um sistema de comunicação, propiciando telemetria e recepção de comandos da estação base; um sistema de posicionamento, utilizando GPS para o traçado de rotas; um sistema de sensoriamento, para leitura da velocidade e direção do vento; um sistema de controle para atuar na variação das angulações das velas e do leme; e um sistema de supervisão, situado na estação-base. Posteriormente, após testarmos o protótipo, um veleiro real será obtido e adaptado, com adição de novos módulos como, por exemplo, sistema de anticolisão, baseado em visão computacional, o sistema meteorológico, para diagnóstico do tempo, o sistema energético, dotado de painéis solares e baterias, o sistema de propulsão motorizado, formado por motor elétrico, e o sistema de limpeza do casco.

Como a plataforma robótica a ser desenvolvida será capaz de atuar em qualquer ambiente aquático, e de forma absolutamente autônoma, é esperada a obtenção de técnicas de controle mais precisas para oferecer uma navegação autônoma confiável, principalmente quando a embarcação utilizar a força dos ventos para se locomover. Estratégias de navegação serão elaboradas considerando-se as correntes das águas, objetivando a economia de energia e melhor desenvolvimento da navegação. Será dada a oportunidade de ser realizada telemetria a grandes distâncias e o envio de missões da estação base.

Este artigo introduz uma proposta de arquitetura de controle de alto nível para barcos robóticos, bem como o contexto no qual a pesquisa está inserida. Em acréscimo a esta introdução, o restante do artigo está estruturado como segue. A seção II apresenta um embasamento teórico, no qual alguns conceitos teóricos importantes à pesquisa são vistos. O estado da arte é apresentado de forma não exaustiva na seção III. Na seção IV o problema abordado é tratado. As implementações realizadas são descritas na seção V. Na seção VI os experimentos e resultados são apresentados. Finalmente, a seção VII conclui o artigo e discute os futuros avanços para a arquitetura de controle proposta.

II. EMBASAMENTO TEÓRICO

A. Navegação a vela

Velejar é uma das mais antigas práticas que o homem desenvolveu para ajudá-lo na sua locomoção e no transporte de carga, a longa distância, utilizando os recursos proporcionados pelo meio aquático e pelo vento. O veleiro é uma máquina utilizada para velejar, que utiliza a força do vento captada pelas velas para movimentá-lo para a frente, através da água.

1) *Partes de um veleiro:* Existem diversos tipos de veleiros, que variam tanto em relação ao formato e tamanho quanto a finalidade. Na Figura 1 encontram-se descritas as partes básicas de um veleiro, cujas definições seguem abaixo.

- **Mastro:** Haste perpendicular ao barco usada para sustentar as velas.
- **Vela mestra:** Ou principal, ou ainda grande, é a maior vela do barco.
- **Buja:** Pequena vela que se localiza na proa do barco.
- **Proa:** Extremidade dianteira do barco.
- **Popa:** Extremidade traseira do barco.

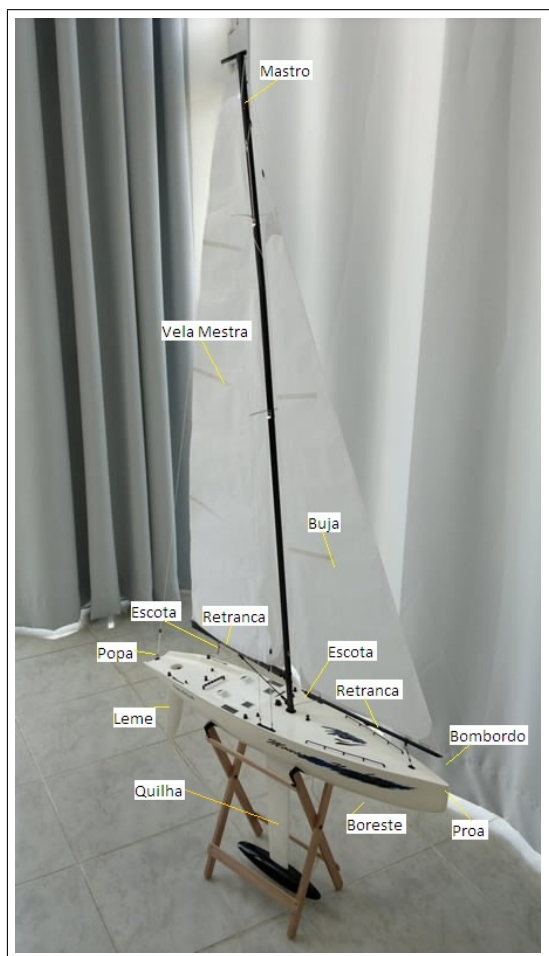


Fig. 1. Partes básicas de um veleiro. [1]

- **Bombordo:** Lado esquerdo do barco.
- **Boreste:** Ou estibordo, é o lado direito do barco.
- **Retranca:** Haste unida à parte inferior da vela.
- **Escota:** Cabo ligado à retranca, cuja função é regular a abertura ou o fechamento das velas.
- **Leme:** Peça laminar utilizada para modificar a direção do barco.
- **Quilha:** Peça em forma de barbatana fixada na parte de baixo do casco do veleiro, cuja função é impedir o abatimento (deslizamento lateral) e auxiliar na estabilidade.

2) *Maneiras de se velejar em relação ao vento:* Segundo [2], dependendo do ângulo formado pelo curso que o veleiro segue e a direção que o vento sopra, o veleiro poderá velejar ou a favor do vento, ou em ângulo reto em relação ao vento ou contra o vento.

No caso de estar velejando a favor do vento, as velas formam em relação a cada lado do barco um ângulo reto para captar a maior quantidade possível de vento. O vento empurra as velas ao encontrá-las, fazendo com que o barco se mova. A velocidade obtida pelo barco é menor do que a do vento devido às perdas ocasionadas pela turbulência gerada pelo movimento do vento em contorna as velas. Essa situação é ilustrada, de forma simplificada, na Figura 2A.

Quando o veleiro está velejando em ângulo reto em relação ao vento, esquema representado na Figura 2B, o vento colide com as velas, que formam um determinado ângulo com um dos lados do barco, e flui paralelamente a estas. Com isso, não ocorre um bloqueio em seu fluxo, como é visto na Figura 2A, e sim um desvio deste. A quilha influe nessa forma de velejar impedindo que o barco se desloque para os lados. Nessa forma de velejar, certos barcos conseguem desenvolver maior velocidade do que o vento.

Na situação de velejar contra o vento, as velas encontram-se posicionadas formando um ângulo menor ainda, em relação a um dos lados do barco, do que quando se veleja em ângulo reto em relação ao vento, fazendo com que o vento se desvie mais ainda. O barco é empurrado para frente, porém com menor velocidade. A quilha aqui é mais solicitada ainda do que na forma de velejar anterior. O esquema é descrito na Figura 2C.

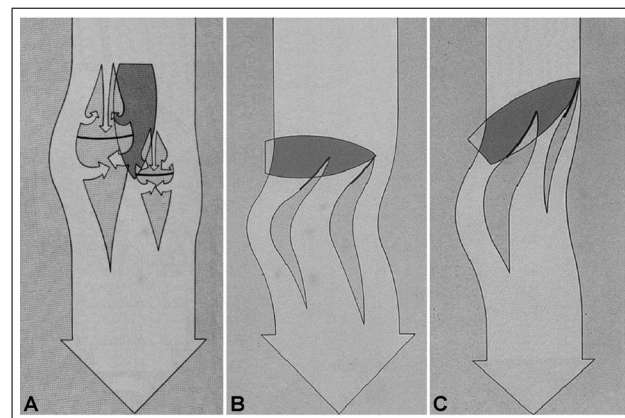


Fig. 2. Maneiras de se velejar em relação ao vento. A) velejando a favor do vento; B) velejando em ângulo reto em relação ao vento; e C) velejando contra o vento. [2]

Na verdade, nenhum barco pode velejar diretamente contra o vento. Há uma região situada a aproximadamente 45 graus de ambos os lados da direção de onde o vento sopra, conhecida como área morta, na qual não é possível atingir determinado ponto em linha reta. Para que esse ponto possa ser atingido, o barco precisa velejar em zigue-zague formando com a direção do vento um ângulo de 45 graus. Essa maneira de velejar recebe o nome de bordejar e cada zigue e cada zague recebe o nome de perna ou bordo. A área morta é ilustrada na Figura 3.

3) *Mareações:* Mareações é o nome que se dá as várias posições que as velas assumem em relação ao vento, podendo possuir diferentes denominações. A Figura 3 apresenta algumas delas.

4) *Mudanças de curso:* Existem quatro tipos básicos de mudança de curso. Quando o barco aproxima a proa da direção do vento, diz-se que o barco está orçando; quando ao contrário, ele afasta a proa da direção do vento, ele está arribando; quanto mais se arriba, mais são folgadas as escotas das velas, e quanto mais se orça mais são caçadas, ou seja, puxadas as escotas das velas.

As velas cruzam o barco de um lado, ou bordo, para o

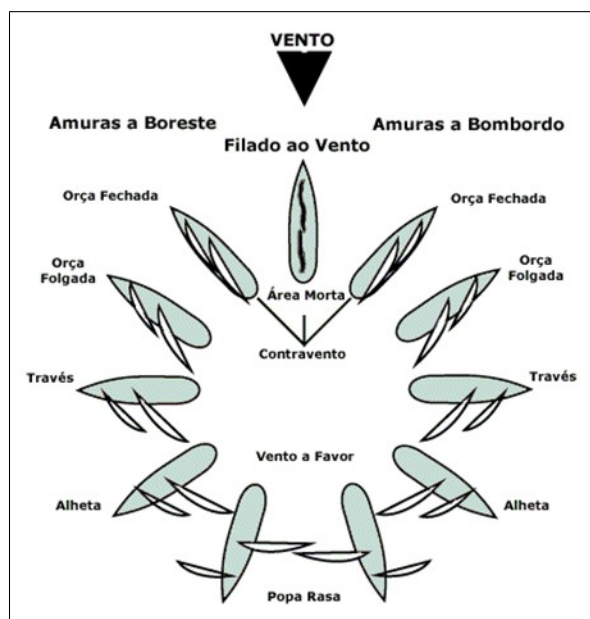


Fig. 3. Mareações que um veleiro pode adotar e localização da área morta. [3]

outro a partir de duas manobras conhecidas por: virar em roda e virar por davante. Virar em roda ou jibar ocorre quando as velas cruzam de um bordo ao outro e o vento incide diretamente na popa do barco. Virar por davante ou bordejar, ou ainda cambiar, ocorre quando as velas cruzam de um bordo ao outro e o vento incide diretamente na proa da embarcação.

B. Instrumentos náuticos

Seguindo a classificação adotada por [4], serão apresentados, de forma não aprofundada, alguns dos instrumentos náuticos utilizados em embarcações.

1) Instrumentos para medida de direção no mar:

- **Agulha eletrônica:** É uma bússola cujo funcionamento se baseia na medida do campo magnético terrestre, apresentando o rumo de forma geralmente digital, não existindo partes móveis em sua constituição, possuindo a capacidade de ser utilizada em latitudes mais elevadas que as bússolas convencionais.
- **Agulha Giroscópica:** É uma bússola precisa e sofisticada que utiliza um giroscópio eletrônico e mancais magnéticos.

2) Instrumentos para medida de velocidade e distância percorrida:

- **Odômetros:** São instrumentos que medem distâncias percorridas e são classificados como: odômetro de superfície, odômetro de fundo e odômetro Doppler. Os odômetros de superfície e de fundo medem a velocidade do navio em relação à massa de água circundante e o odômetro Doppler tem a capacidade de medir a velocidade do navio em relação ao fundo do mar, sendo que a distância percorrida é obtida pela integração da velocidade em relação ao tempo por meios elétricos ou mecânicos.

- **Velocímetros:** São instrumentos que medem a velocidade da embarcação em relação à água e são geralmente atuados por uma haste submersa que é pressionada para trás quando a embarcação se encontra em movimento. O velocímetro hidráulico e o de hélice são exemplos deste tipo de instrumento.
- **GPS:** Segundo [4], o Sistema de Posicionamento Global (*Global Positioning System*), abreviadamente, GPS, é formado por três partes principais: satélites; sistema de monitoramento e controle; e aparelhos receptores GPS e equipamentos de apoio. As três partes operam em conjunto, fornecendo latitude, longitude, altitude, rumo, velocidade e hora, com alta precisão.

3) Instrumentos para medida de profundidades:

- **Ecobatímetro:** Também conhecido como sonda sonora, é um dispositivo que utiliza feixes de ondas sonoras ou ultra-sonoras para medir a profundidade em que se encontra a embarcação. Conhecendo a velocidade do som na água, basta medir o tempo decorrido entre a emissão do sinal e a recepção do eco refletido do fundo para se obter a profundidade.

III. ESTADO DA ARTE

Dos projetos de veleiros autônomos pesquisados, três se destacam pelos seus desempenhos no campo prático e pela quantidade de material de pesquisa publicada. São eles, o *ASV Roboat*, o *FASt*, e o *Avalon*.

A. ASV Roboat

O veleiro autônomo ASV Roboat foi construído, como segundo protótipo pela equipe da INNOC, para participar do Desafio Microtransat¹ de 2007 ocorrido no mar da Irlanda [5].

Seu *software* de controle é executado numa distribuição do sistema operacional Linux no computador de bordo e está conectado aos sensores e aos atuadores. O sistema de sensoriamento é parcialmente redundante e mede as seguintes grandezas: velocidade e direção do vento, direção do barco, velocidade angular, adernamento, posição do barco, velocidade do barco, profundidade da água, temperatura da água, temperatura do ar, umidade do ar e pressão atmosférica. O veleiro é equipado com atuadores para as velas, leme, luz de navegação, buzina de nevoeiro e bomba de porão.

O ASV Roboat possui duas fontes de energia: a principal, que são os painéis solares que podem fornecer até 285 Wp, que abastecem um banco de baterias de 1,92 kWh; e a de emergência que é constituída de uma célula combustível de metanol que fornece 65 W.

A arquitetura do sistema é composta por duas partes principais: a arquitetura de subsunção híbrida em quatro camadas (Figura 4) e o sistema de comunicação em três estágios (Figura 5).

A arquitetura de subsunção híbrida combina as abordagens deliberativa e a reativa e é constituída por quatro camadas: roteamento estratégico de longo prazo, roteamento de curso

¹ <http://www.microtransat.org>

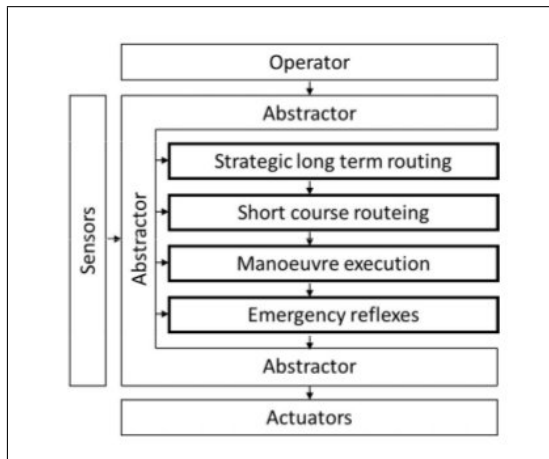


Fig. 4. Diagrama da arquitetura de subsunção híbrida em quatro camadas do ASV Roboat. [5]

curto, execução de manobra e reflexos de emergência. Elas são executadas de forma assíncrona e concorrente. Cada camada acessa os dados oriundos dos sensores mediante o abstrator, que é um programa que roda no computador de bordo do veleiro. Esse programa é responsável por processar os dados brutos entregues pelos sensores, transformando-os em dados inteligíveis para as camadas.

- **Roteamento estratégico de longo prazo:** É executado o algoritmo de roteamento que define uma rota ótima tendo como base determinados parâmetros, como, por exemplo, as condições do tempo e a topologia do mar. A rota é dividida em várias sub-rotas e as coordenadas dos pontos de união dessas sub-rotas são passadas sequencialmente para a camada de roteamento de curso curto.
- **Roteamento de curso curto:** Determina um caminho ótimo para atingir o ponto alvo dado pela camada de roteamento estratégico de longo prazo. São levados em consideração para a escolha do melhor caminho as condições do vento e o tipo de maré a ser seguido.
- **Execução de manobra:** Recebe da camada de roteamento de curso curto uma direção desejada que o veleiro deve seguir. Caso ocorra desvio do curso, o sistema pertencente a essa camada ajusta a posição do leme para colocar o barco novamente no curso desejado. Um segundo sistema é executado em paralelo para garantir fluxo de ar nas velas para que possa haver propulsão. Esta camada tem de bordejar ou jibar caso seja preciso ao longo da navegação.
- **Reflexos de emergência:** Recebe da camada de execução de manobra as posições desejadas do leme e das velas. Essas posições são repassadas inalteradas aos atuadores quando não existe uma ameaça à segurança do barco. Caso exista, as ações solicitadas são ignoradas.

O sistema de comunicação de três estágios oferece os recursos necessários para haver a troca de dados entre o veleiro, o *software* de visualização e o controlador remoto. Ocorre chaveamento dinâmico automático entre os canais de comunicação disponíveis no veleiro. Os estágios são os

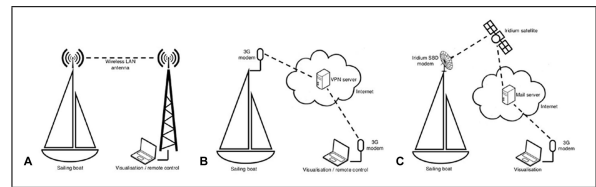


Fig. 5. Sistema de comunicação de três estágios do ASV Roboat. A) Primeiro estágio; B) Segundo estágio; C) Terceiro estágio. [5]

seguintes:

- **Primeiro estágio:** *wireless* LAN;
- **Segundo estágio:** serviço de dados 3G de provedor de telefonia móvel;
- **Terceiro estágio:** comunicação via satélite (modem IRIDIUM 9601 SBD).

B. FAST

O projeto FAST [6] foi lançado pelo Departamento de Elétrica e de Computação da Faculdade de Engenharia da Universidade do Porto (FEUP), Portugal, em 2007, tendo como primeiro objetivo participar das competições Microtransat.

Seu sistema eletrônico é organizado em cinco subsistemas principais: computação, comunicação, sensores, atuadores e gerenciamento de energia. O sistema de computação é implementado em um pequeno computador de bordo baseado em um FPGA (*Field-Programmable Gate Array*) incluindo um microprocessador RISC, que fornece uma plataforma flexível de computação. O sistema de comunicação utiliza um roteador WiFi, para comunicação de curto alcance com um computador pessoal, um modem GSM, para comunicações de pequenos volumes de dados no mar a poucos quilômetros da costa, um modem IRIDIUM SBD, para comunicações a longas distâncias no mar em escala mundial, e um receptor de rádio controle, para controle totalmente manual da embarcação. Todos estes componentes são integrados como módulos de OEM e podem ser ligados e desligados sob o controle do *software* em execução no sistema de computação. Os sensores utilizados são: anemômetro, biruta, posicionador da retranca, bússola digital, GPS, inclinômetro, monitores de voltagem, sensor de luz ambiente, termômetro e um conjunto de sensores de água. O veleiro inclui três atuadores: dois servos RC de alta potência que controlam de forma independente dois lemes e um motor DC que controla as velas buja e a mestra simultaneamente. O subsistema de gerenciamento de energia inclui um painel solar de 45 Wp, duas baterias de íon-lítio de 95 Wh, um carregador de bateria, um módulo de gerenciamento e uma fonte de alimentação ultra eficiente.

O sistema operacional utilizado pelo FAST, em seu computador de bordo, é o μ CLinux², que é uma versão do sistema operacional Linux voltada para microcontroladores sem MMU. Ele fornece um console interativo de linha de comando, por meio de uma porta RS232 padrão, um

²www.uclinux.org

sistema de arquivo estruturado, multitarefa e serviços TCP/IP básicos, como FTP, HTTP e TELNET.

Os dados coletados pelos sensores são entregues ao *software* de controle por meio de interfaces autônomas, após serem tratados. O *software* de controle é constituído por um conjunto de processos que se comunicam por *sockets* UDP. A sua arquitetura tenta imitar a sequência de comandos existente nas operações de barcos reais, que seguem uma hierarquia. Os comandos transitam do nível superior para o inferior como pacotes UDP.

O *software* é composto por cinco aplicativos: interface de *hardware*, *helm*, *sail*, *skipper* e *logger*. A interface de *hardware* implementa toda a comunicação com os sensores, atuadores e parâmetros de configuração. O *helm* controla os lemes, segundo algoritmos de controle e comandos providos do *skipper*. Manter um curso e executar as manobras básicas de bordejo e jibe, são exemplos desses comandos. O *sail* controla a posição das velas, com base na velocidade e direção do vento e um conjunto de regras que estabelecem a melhor angulação para estas. O *skipper* é o processo que ocupa o nível superior. Pela interface de *hardware* ele tem acesso a todas as informações coletadas pelos sensores. O *skipper* não tem ação direta sobre os atuadores, ficando esta sob a responsabilidade do *helm*, para os lemes, e o *sail*, para as velas. Ele também é responsável por decidir qual rumo tomar para chegar a um determinado ponto alvo, quando bordejar ou jibar e comutar para o modo de controle manual ou autônomo. O *logger* é responsável por gravar todos os dados relevantes colhidos dos outros processos em um arquivo *log*.

C. Avalon

O veleiro autônomo Avalon foi construído pela ETH-Zurique com o objetivo de cumprir o desafio principal da competição Microtransat, que é a travessia do Oceano Atlântico de forma totalmente autônoma [7].

O Avalon utiliza como computador de bordo um MPC21 da DigitalLogic com 500 MHz, 1 GB de RAM e um disco rígido flash compacto. O veleiro é equipado com um GPS, uma unidade de medida inercial, um sensor de vento de ultra-som, que fornece a velocidade e direção do vento, e um sensor AIS (*Automatic Identification System*), que recebe como dados a posição e velocidade de outros barcos em VHF, e é utilizado no sistema de anti-colisão. Para comunicação via satélite o veleiro utiliza um modem IRIDIUM 9522-B. A energia elétrica provem de um banco de baterias de lítio-manganês com capacidade total de 2,4 KWh alimentado por quatro painéis solares que produzem 90 Wp cada um. Uma célula de combustível de metanol é utilizada como fonte de energia de emergência. O veleiro utiliza um motor de 200 W para o controle da vela e dois motores de 150 W para controlar os lemes [8].

A estrutura do *software* do Avalon se baseia em uma *Dynamic Data eXchange* (DDX), que é executada no Debian, uma distribuição do sistema operacional Linux. Esse *software* gerencia uma memória compartilhada chamada *store*

que propociona um modo de comunicação entre os subprogramas, que o constitui, e que são executados em forma concorrente (Figura 6). Os *drivers* dos sensores coletam os dados lidos e os armazenam na memória compartilhada, na qual podem ser utilizados pelos programas de controle e de planejamento de curso [7].

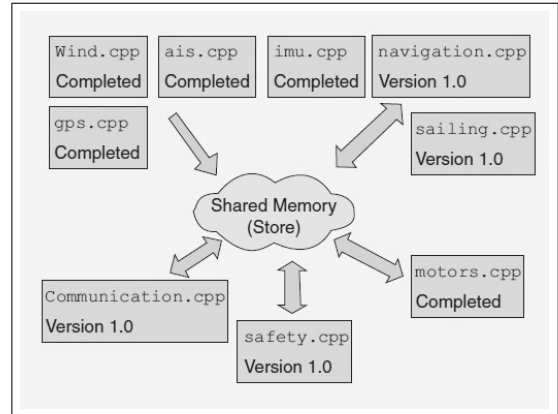


Fig. 6. Organização do *software* do Avalon. [7]

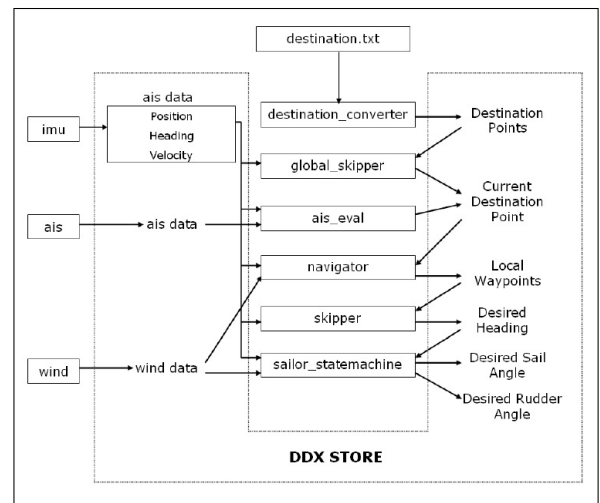


Fig. 7. Diagrama da arquitetura do *software* do Avalon. [9]

A arquitetura do *software* pode ser visualizada com detalhes na Figura 7. Os sensores utilizados pelo Avalon são: IMU, AIS e o WIND. A Unidade de Medidas Inercial (IMU) utilizada é, na realidade, uma combinação de uma IMU, um GPS e um barômetro e fornece como dados a posição, velocidade e aceleração, além da orientação e velocidade angular em três dimensões. O AIS fornece informações referentes a outras embarcações que estão navegando próximo ao veleiro. O sensor de vento (WIND) fornece a velocidade e direção do vento. O *destination_converter* tem como tarefa ler todos os pontos de destino contidos no arquivo texto *destination.txt* e escrevê-los na memória compartilhada. O *global_skipper* calcula o ponto de destino atual com base nas informações obtidas do IMU e nos pontos de destino armazenados. O *ais_eval* avisa ao *global_skipper* quando há a possibilidade de ocorrer uma colisão, fazendo com que ele calcule outro ponto

de destino atual. O *navigator* calcula o melhor caminho entre a posição corrente e o ponto de destino atual para que o Avalon possa seguir. Ele gera diferentes *waypoints* e os armazena na memória compartilhada juntamente com os ângulos formados entre os *waypoints* consecutivos. O *skipper* gerencia os *waypoints* e decide qual posição desejada o Avalon precisa seguir, armazenando-a na memória compartilhada. O *sailor_statemachine* calcula os ângulos desejados dos lemes e da vela a partir da velocidade, posição atual e da posição desejada [9].

IV. O PROBLEMA E A SOLUÇÃO

O problema aqui enfrentado é o desenvolvimento de uma arquitetura de controle de alto nível para a plataforma N-Boat, de modo a torná-la totalmente autônoma em suas missões. Tal arquitetura de controle tem de gerenciar todos os módulos existentes no sistema, de forma a capacitar a plataforma a tomar decisões autonomamente. Do que foi dito, é esperado que a plataforma, quando em missão, determine o melhor rumo a ser tomado, tendo como base um ponto alvo, fornecido pela estação-base, e os dados coletados pelos seus sensores. Em sequência, escolher a melhor estratégia de navegação para seguir, bordejando ou jibando quando necessário, e sempre estar ciente da situação de sua segurança. Caso venha ocorrer algo que a comprometa, ter a habilidade de contornar o problema e retornar a um estado seguro de navegação.

No projeto da plataforma N-Boat é previsto equipá-la com os componentes apresentados na Tabela I.

TABELA I
ESTIMATIVA DE COMPONENTES DO N-BOAT.

Item	Componente
01	Propulsão a vento
02	Células fotovoltaicas / Painéis solares
03	Receptor GPS / Sensor de nuvens / Estação meteorológica
04	Payload de parceiros / Instrumentos científicos
05	Banco de baterias / Sistema de potência
06	Servos de dois graus de liberdade do controle de vela
07	Computador de bordo
08	Comunicação (WiFi, VHF, Satélite, Rádio-modem, 3G, GPRS)
09	Bomba de drenagem de água
10	Propulsão elétrica alternativa
11	Servo do leme
12	Câmera estéreo
13	Laserscan / Ultrasom / Sensor de pressão
14	Sirene / Outros instrumentos / IMU (unidade de medidas inerciais)

Tendo como ponto de partida o conhecimento dos módulos mínimos necessários para tornar uma plataforma totalmente autônoma e os prováveis componentes que estarão a bordo do N-Boat, é sugerida a arquitetura de controle apresentada na Figura 8 para a plataforma e na Figura 9 para a estação-base.

O *módulo de sensoramento* capta todas as informações do ambiente necessárias à plataforma e as escreve no módulo de memória. O *módulo de comunicação* permite que a

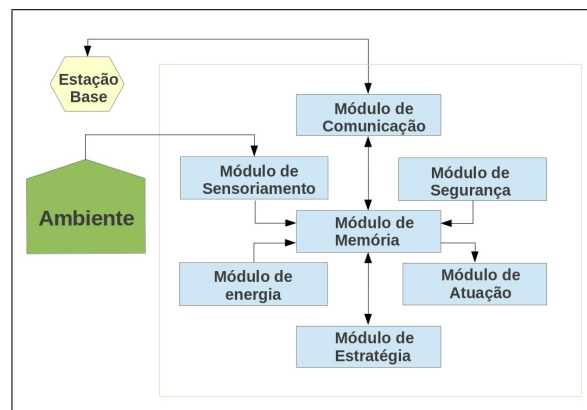


Fig. 8. Diagrama da arquitetura de controle proposta (plataforma).

plataforma troque dados com a estação-base, realizando telemetria e recebendo missões. Esse módulo tanto pode escrever quanto ler no módulo de memória. O *módulo de energia* monitora os painéis solares e as baterias, podendo escrever no módulo de memória. O *módulo de segurança* é constituído por um sistema de anti-colisão e um sistema de detecção de mal tempo objetivando proporcionar ao N-Boat uma navegação segura. Esse módulo pode escrever no módulo de memória. Ambos os sistemas possuem recursos baseados em visão computacional. O *módulo de estratégia* é o mais importante. Ele é responsável pela determinação do melhor rumo que a plataforma deve tomar, bem como a escolha da melhor manobra a ser seguida. É capaz de planejar a execução das manobras de bordejo e de jibe, e lê e escreve no módulo de memória. O *módulo de atuação* realiza o controle das velas e do leme. Também é responsável pelo controle dos componentes relacionados à propulsão elétrica alternativa. O *módulo de memória* é responsável pelo tratamento, armazenamento e gerenciamento de todos os dados providos dos outros módulos, permitindo a eles as operações de leitura e/ou escrita.

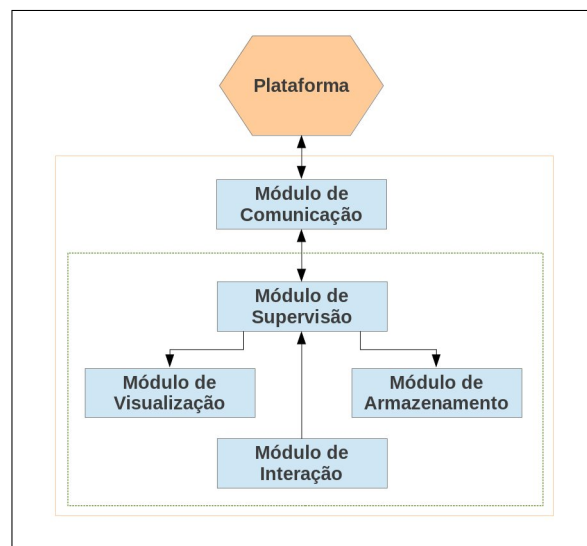


Fig. 9. Diagrama da arquitetura de controle proposta (estação-base).

Como ocorre na arquitetura da plataforma, o *módulo de comunicação* da arquitetura da estação-base permite que esta troque dados com a plataforma, recebendo telemetria e enviando missões. Os módulos de *supervisão*, *visualização*, *interação* e *armazenamento* funcionam em conjunto e dão todo o suporte que um sistema supervisorio oferece, ou seja, apresentar os dados recebidos via telemetria, armazená-los e permitir interação com a plataforma, mediante envio de dados pelo módulo de comunicação.

V. IMPLEMENTAÇÕES

A arquitetura proposta neste trabalho foi implementada, de forma simplificada, em um simulador desenvolvido na linguagem C e executado no sistema operacional Linux.

O funcionamento de cada módulo presente na arquitetura de controle da plataforma, excetuando o módulo de segurança, foi simulado em um *software*. O mesmo foi feito com os módulos de comunicação e supervisão da arquitetura de controle da estação-base.

Para gerenciar a arquitetura da plataforma foi implementado um algoritmo de controle e para gerar as rotas foi implementado um algoritmo de navegação. Ambos os algoritmos são apresentados a seguir.

A. Algoritmo de controle

O pseudo-código do algoritmo de controle é apresentado a seguir:

```

01 Início
02   Executa(módulo_de_memória)
03   Executa(módulo_de_sensoriamento)
04   Executa(módulo_de_comunicação)
05   Executa(módulo_de_estratégia)
06   Executa(módulo_de_atuação)
07   Executa(módulo_de_energia)
08
09   while(1){
10     Escreve_Lê(módulo_de_sensoriamento,
11               módulo_de_comunicação,
12               módulo_de_estratégia,
13               módulo_de_atuação,
14               módulo_de_energia)
15   }
16 Fim

```

Basicamente ele segue os seguintes passos:

- 1) O *software* do módulo de memória é executado (linha 2).
- 2) Os *softwares* dos demais módulos são executados, sendo a ordem apresentada não obrigatória (linhas 3 a 7).
- 3) Cada módulo lê e/ou escreve de forma concorrente na memória compartilhada, sendo o seu acesso controlado por um semáforo binário (linha 10).

B. Algoritmo de navegação

O pseudo-código do algoritmo de navegação é apresentado a seguir:

```

01 Início
02   while(1){
03     Lê(coordenadas_do_barco)

```

```

04     Lê(coordenadas_do_alvo)
05     Calcula(distância)
06     Calcula(direção_alvo)
07     Lê(direção_vento)
08     d <- |direção_alvo - direção_vento|
09
10     if(d>=0.E.d<=45)
11       f <- 45 - d
12     else if(d>45.E.d<=150)
13       f <- 0
14     else
15       f <- d - 150
16
17     quant <- Calcula(pontos_de_percurso)
18     contador <- 1
19
20     while(contador < quant){
21       f <- -f
22       Lê(direção_barco)
23
24       do{
25         if(direção_barco <> direção_alvo + f){
26           Move_velas_leme(direção_alvo + f)
27         }
28
29         Lê(direção_barco)
30         Lê(coordenadas_do_barco)
31       }while(coordenadas_do_barco <>
32             pontos_de_percurso[contador])
33
34       contador <- contador + 1
35     }
36   }
37 Fim

```

Os passos mais importantes do algoritmo são:

- 1) A diferença das direções do alvo e do vento é calculada (linha 8).
- 2) É verificado se a diferença está entre 0 e 45 graus, que caracteriza a navegação contra o vento (linha 10).
- 3) Nas linhas 12 e 14 são tratados os casos de navegação em curso reto e a favor do vento, respectivamente.
- 4) Os pontos de percurso são calculados (linha 17).
- 5) Na linha 26 é executado o ajuste nas velas e no leme.

VI. EXPERIMENTOS E RESULTADOS

O simulador implementado é constituído de processos individuais que se comunicam mediante uma memória compartilhada controlada por um semáforo binário. A memória compartilhada foi implementada no módulo de memória juntamente com o algoritmo de controle. O algoritmo de navegação foi implementado no módulo de estratégia.

As Figuras 10, 11 e 12 apresentam os gráficos traçados pelo Scilab das coordenadas geradas pelo simulador dos pontos de percurso nos casos de navegação contra o vento, navegação em linha reta e navegação a favor do vento, respectivamente. Nessa simulação, foram consideradas como coordenadas do ponto atual do barco e do alvo os valores lat. 34°3'8"N lon. 118°14'37"W e lat. 34°3'58"N lon. 118°16'11"W, respectivamente.

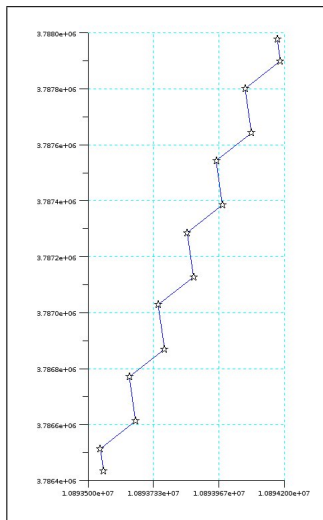


Fig. 10. Rota gerada pelo simulador. Direção do vento-norte: $37,00^\circ$; direção do alvo-norte: $21,87^\circ$; quantidade de subpercurso: 12; distância em linha reta: 1664,30 m; distância entre subpercurso: 138,69 m; distância percorrida entre subpercurso: 159,94 m; distância total estimada: 1919,28 m; afastamento da linha de rumo: 39,83 m; situação: brodejando.

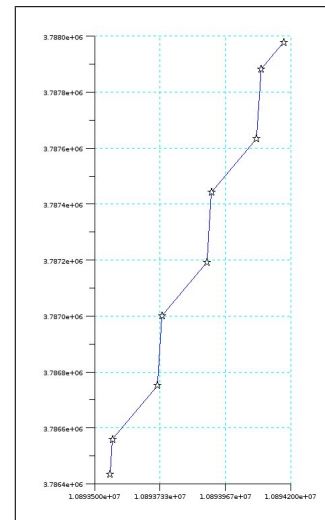


Fig. 12. Rota gerada pelo simulador. Direção do vento-norte: $-170,00^\circ$; direção do alvo-norte: $21,87^\circ$; quantidade de subpercurso: 7; distância em linha reta: 1664,30 m; distância entre subpercurso: 237,76 m; distância percorrida entre subpercurso: 250,17 m; distância total estimada: 1751,19 m; afastamento da linha de rumo: 38,92 m; situação: jibando.

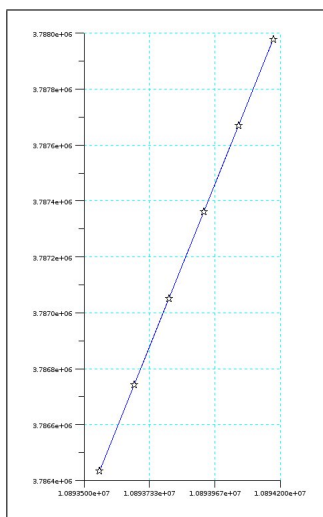


Fig. 11. Rota gerada pelo simulador. Direção do vento-norte: $150,00^\circ$; direção do alvo-norte: $21,87^\circ$; quantidade de subpercurso: 5; distância em linha reta: 1664,30 m; distância entre subpercurso: 332,86 m; distância percorrida entre subpercurso: 332,86 m; distância total estimada: 1664,30 m; afastamento da linha de rumo: 0,0 m; situação: em linha reta.

VII. CONCLUSÕES

Os experimentos descritos neste artigo tiveram como foco a implementação simplificada de um simulador, objetivando o desenvolvimento de uma arquitetura de controle de alto nível para plataformas robóticas aquáticas autônomas.

Após cumprir todos os requisitos de funcionamento e ser devidamente testada, tal arquitetura de controle será embarcada em *hardware* para controlar o protótipo e, em seguida, após as depurações exigidas, ser embarcada no sistema do veleiro N-Boat a ser adquirido.

As pesquisas referentes aos módulos de comunicação, sensoramento e atuação já se encontram em andamento sob a responsabilidade dos demais membros da equipe do projeto

N-Boat, do laboratório Natalnet.

AGRADECIMENTOS

Os autores agradecem ao CNPq (Projeto Universal 485706/2012-1) e à CAPES, pelo apoio financeiro e bolsa.

REFERÊNCIAS

- [1] Hobbycontrol (2013), Página na Internet. <http://www.hobbycontrol.com.br/details/CAISER-01/monsoon-900-branco>
- [2] N. Dent (1978), Velejar – Manual prático de técnicas e equipamentos, Editora Abril.
- [3] Portal São Francisco (2013), Página na Internet. <http://www.portalsaofrancisco.com.br/alfa/barco-a-vela/barco-a-vela-5.php>
- [4] A. P. Miguens (2000), Navegação: A Ciência e a Arte, Marinha do Brasil.
- [5] R. Stelzer (2012), Autonomous Sailboat Navigation – Novel Algorithms and Experimental Demonstration, Tese de doutorado, De Montfort University, Leicester.
- [6] J. C. Alves & N. A. Cruz (2008), 'Fast – an autonomous sailing platform for oceanographic missions', IEEE.
- [7] Erckens, Hendrik, G. Büsser, C. Pradalier & R. Y. Siegwart (2010), 'Avalon – navigation strategy and trajectory following controller for an autonomous sailing vessel', IEEE .
- [8] C. Alt & N. Wittinghofer (2011), Autonomous sailing boats, Relatório técnico, Universität Salzburg, Áustria.
- [9] M. Krucker (2010), Simulation, control and testing of an autonomous sailboat, Dissertação de mestrado.