Como Ler um Artigo Científico

Prof: Marcia Pessini

O que é um artigo científico?

- Apresentação sucinta de uma investigação científica
  - Normalmente submetido a exame e validação por parte de outros cientistas.
  - Escrito segundo uma linguagem e metodologias próprias de uma dada área da ciência.
  - Tipicamente apresentado com uma estrutura lógica de argumentação semelhante.

Qual a importância do artigo científico?

- É a base da divulgação do conhecimento científico
  - A ciência é uma forma de conhecimento de carácter público, cuja validade só se estabelece após o debate em torno dos resultados apresentados e do caminho percorrido que conduziu à sua construção.
  - O artigo científico torna público e aberto ao debate, o conhecimento resultante da investigação, sendo por isso o motor da divulgação e do desenvolvimento da ciência.
- É a base da avaliação académica (pessoal e institucional)

Motivação para ler artigos

- Bibliografia na realização de trabalhos acadêmicos
- Conhecer o “estado da arte” de uma área
- Manter-se atualizado
- Seguir o trabalho científico de outros
- Estudar metodologias
- Comparar resultados
- Encontrar referências para fundamentar a sua investigação
- Aprender sobre “escrita técnica e científica”
- . . .

Motivação para uma metodologia

- Por vezes os artigos são difíceis de ler
  - Escritos num estilo muito condensado devido a restrições de formato e número de páginas.
  - Escritos numa linguagem muito técnica pois são dirigidos a *experts*.
  - Não são especialmente escritos para encontrarmos facilmente o que procuramos.
  - Não seguem os objetivos da escrita técnica e científica que deve ser objectiva e clara.

- Facilmente nos perdemos na imensidão de artigos disponíveis

- É irrealista (e inútil) ler tudo o que existe sobre um tema
  - A procura deve ser exaustiva, mas a leitura deve ser seletiva.
  - A sobrecarga de informação pode confundir.
  - Começar com resumos e depois escolha o que lhe parecer mais interessante.
  - Identifique sempre as suas fontes de informação.
  - Anote a informação que quer retirar do artigo.
  - Registe as suas referências bibliográficas e construa com elas um arquivo.

Metodologia

- Que informação queremos retirar do artigo?
  - Quais são as questões a que queremos responder?

- Fato: Os artigos têm uma estrutura muito semelhante e repetitiva.
  - Facilita a leitura e a procura de informação.

Estrutura típica de um artigo

- **Cabeçalho**
  - Título do artigo e identificação dos autores com os seus contatos e filiações.
- **Resumo**
  - Parágrafo que resume o artigo fazendo referência ao problema, à solução e aos resultados.
- **Introdução**
  - Motivação do trabalho, definição do problema, visão geral da solução e descrição da estrutura do artigo.
- **Estado da arte**

Questões a responder

- Quais são as motivações do trabalho?
- Qual é a solução proposta?
- Como é que a solução foi avaliada?
- Qual é a sua análise crítica sobre o problema, a solução e a avaliação?
- Quais são as contribuições do trabalho apresentado?
- Quais são as direções futuras da investigação?
- Quais são as questões que surgiram?
- Qual é a mensagem a reter do artigo?
- Quais são as referências que podem ser úteis no futuro?
- Quais são os elementos da escrita que podemos aproveitar?

Questões a responder

- Quais são as motivações do trabalho?
  - Resolver um problema para o qual ainda não se publicou uma solução.
    - Porque é que o problema não tem uma solução trivial?
  - Encontrar uma solução mais adequada para um problema.
    - Que soluções existem e porque não são adequadas?
  - Rever uma área de conhecimento.
  - Qual é o problema do mundo real e qual é o problema técnico?

Questões a responder

- Qual é a solução (método, sistema, ideia, hipótese, …) proposta?
  - Porque é que se assume que a solução funcionará?
  - Porque é que a solução é melhor do que outras já existentes?
  - Como é que a solução será desenvolvida e implementada?

Questões a responder

- Como é que a solução foi avaliada?
  - Uma simples ideia não é normalmente suficiente para publicar num artigo científico!
  - Que argumentos, experiências, implementações (protótipos) e resultados são descritos que avaliam a qualidade da solução?

Questões a responder

- Qual é a sua análise crítica sobre o problema, a solução e a avaliação?
  - O problema é relevante, faz sentido?
  - A solução foi bem testada/avaliada?
  - Que falhas podemos identificar no trabalho?
  - Quais são os pontos fortes do trabalho?
  - Quais são os pontos fracos do trabalho?
  - A ideia vai mesmo funcionar, alguém a vai querer, é implementável?

Questões a responder

- Quais são as contribuições do trabalho apresentado?
  - Na opinião dos autores e na sua opinião.
  - As contribuições de um artigo podem ser muitas e variadas:
    - Novos conhecimentos
    - Ideias
    - Sistemas
    - Técnicas
    - Estado da arte
    - etc

Questões a responder

- Quais são as direções futuras da investigação?
  - Quais as direções futuras identificadas pelos autores?
  - Que novas ideias lhe surgiram depois de ler o artigo?

Questões a responder

- Quais são as questões que surgiram?
  - Que questões gostaria de debater sobre o artigo?
  - O que achou confuso ou difícil de entender?

Questões a responder

- Qual é a mensagem a reter do artigo?
  - Use a resposta para no futuro refrescar a memória sobre o conteúdo do artigo.

Questões a responder

- Quais são as referências que podem ser úteis no futuro?
  - Quais são os artigos referenciados que acha que vale a pena ler assim que possível e os que podem ser úteis no futuro.

Questões a responder

- Quais são os elementos da escrita que podemos aproveitar?
  - Como é que a informação foi organizada ao longo do artigo?
  - Como é que certos aspectos da investigação foram descritos por palavras e por imagens?
  - Que métodos foram usados para apresentar e comparar os resultados?

Onde procurar as respostas

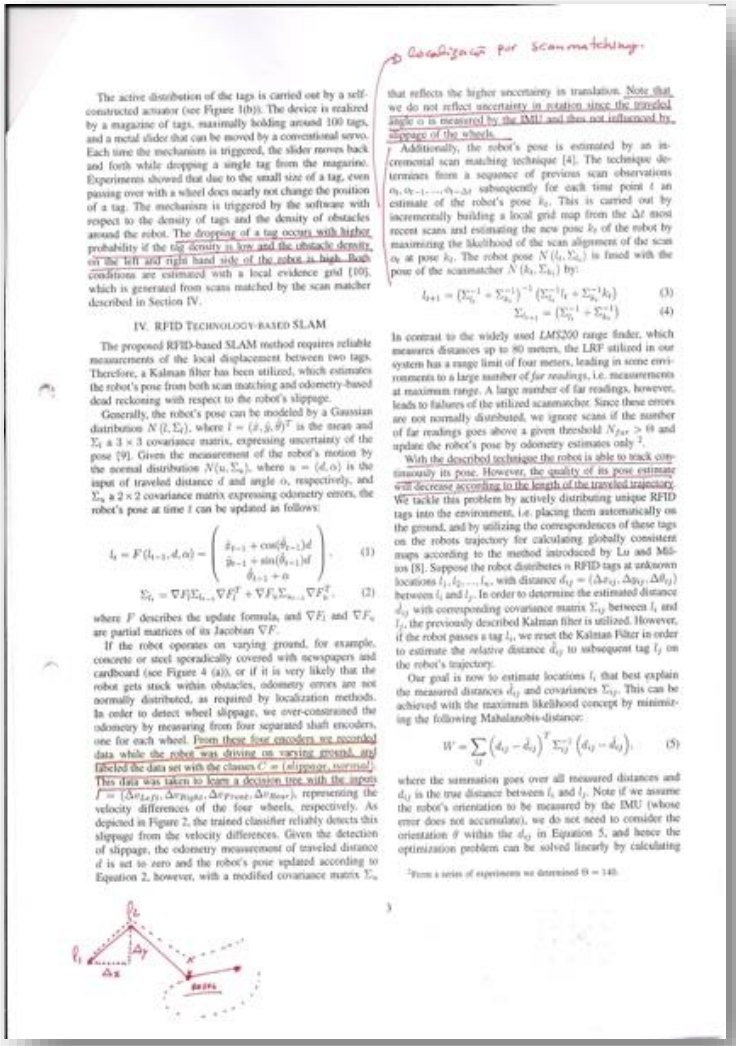
- A motivação estará normalmente na introdução.
- Uma descrição genérica da solução e da avaliação aparece normalmente na introdução e nas conclusões.
- Uma descrição mais detalhada estará no corpo do artigo.
- O trabalho futuro costuma aparecer nas conclusões, juntamente com o sumário das contribuições.

Onde anotar as respostas

- No próprio artigo

- Num formulário próprio

Anotações no próprio artigo



The active distribution of the tags is carried out by a self-constructed actuator (see Figure 1(b)). The device is realized by a magazine of tags, maximally holding around 100 tags, and a metal slider that can be moved by a conventional servo. Each time the mechanism is triggered, the slider moves back and forth while dropping a single tag from the magazine. Experiments showed that due to the small size of a tag, even passing over with a wheel does nearly not change the position of a tag. The mechanism is triggered by the software with respect to the density of tags and the density of obstacles around the robot. The dropping of a tag occurs with higher probability if the tag density is low and the obstacle density on the left and right hand side of the robot is high. Both conditions are estimated with a local evidence grid [10], which is generated from scans matched by the scan matcher described in Section IV.

IV. RFID Technology-based SLAM

The proposed RFID-based SLAM method requires reliable measurements of the local displacement between two tags. Therefore, a Kalman filter has been utilized, which estimates the robot's pose from both scan matching and odometry-based dead reckoning with respect to the robot's slippage.

Generally, the robot's pose can be modeled by a Gaussian distribution $N(\hat{l}, \Sigma_l)$, where $\hat{l} = (\hat{x}, \hat{y}, \hat{\theta})^T$ is the mean and Σ_l a 3×3 covariance matrix, expressing uncertainty of the pose [9]. Given the measurement of the robot's motion by the normal distribution $N(u, \Sigma_u)$, where $u = (d, \alpha)$ is the input of traveled distance d and angle α, respectively, and Σ_u a 2×2 covariance matrix expressing odometry errors, the robot's pose at time t can be updated as follows:

$$l_t = F(l_{t-1}, d, \alpha) = \begin{pmatrix} \hat{x}_{t-1} + \cos(\hat{\theta}_{t-1})d \\ \hat{y}_{t-1} + \sin(\hat{\theta}_{t-1})d \\ \hat{\theta}_{t-1} + \alpha \end{pmatrix}, \quad (1)$$

$$\Sigma_{l_t} = \nabla F_l \Sigma_{l_{t-1}} \nabla F_l^T + \nabla F_u \Sigma_{u_{t-1}} \nabla F_u^T, \quad (2)$$

where F describes the update formula, and ∇F_l and ∇F_u are partial matrices of its Jacobian ∇F.

If the robot operates on varying ground, for example, concrete or steel sporadically covered with newspapers and cardboard (see Figure 4 (a)), or if it is very likely that the robot gets stuck within obstacles, odometry errors are not normally distributed, as required by localization methods. In order to detect wheel slippage, we over-constrained the odometry by measuring from four separated shaft encoders, one for each wheel. From these four encoders we recorded data while the robot was driving on varying ground, and labeled the data set with the classes $C = \{slippage, normal\}$. This data was taken to learn a decision tree with the inputs $f = (\Delta v_{Left}, \Delta v_{Right}, \Delta v_{Front}, \Delta v_{Rear})$, representing the velocity differences of the four wheels, respectively. As depicted in Figure 2, the trained classifier reliably detects this slippage from the velocity differences. Given the detection of slippage, the odometry measurement of traveled distance d is set to zero and the robot's pose updated according to Equation 2, however, with a modified covariance matrix Σ_u that reflects the higher uncertainty in translation. Note that we do not reflect uncertainty in rotation since the traveled angle α is measured by the IMU and thus not influenced by slippage of the wheels.

Additionally, the robot's pose is estimated by an incremental scan matching technique [4]. The technique determines from a sequence of previous scan observations $o_t, o_{t-1}, ..., o_{t-\Delta t}$ subsequently for each time point t an estimate of the robot's pose $\hat{k}_t$. This is carried out by incrementally building a local grid map from the Δt most recent scans and estimating the new pose k_t of the robot by maximizing the likelihood of the scan alignment of the scan o_t at pose k_t. The robot pose $N(\hat{l}_t, \Sigma_{l_t})$ is fused with the pose of the scanmatcher $N(k_t, \Sigma_{k_t})$ by:

$$\hat{l}_{t+1} = \left(\Sigma_{l_t}^{-1} + \Sigma_{k_t}^{-1}\right)^{-1} \left(\Sigma_{l_t}^{-1}\hat{l}_t + \Sigma_{k_t}^{-1}k_t\right) \quad (3)$$

$$\Sigma_{l_{t+1}} = \left(\Sigma_{l_t}^{-1} + \Sigma_{k_t}^{-1}\right) \quad (4)$$

In contrast to the widely used *LMS200* range finder, which measures distances up to 80 meters, the LRF utilized in our system has a range limit of four meters, leading in some environments to a large number of *far readings*, i.e. measurements at maximum range. A large number of far readings, however, leads to failures of the utilized scanmatcher. Since these errors are not normally distributed, we ignore scans if the number of far readings goes above a given threshold $N_{far} > \Theta$ and update the robot's pose by odometry estimates only [2].

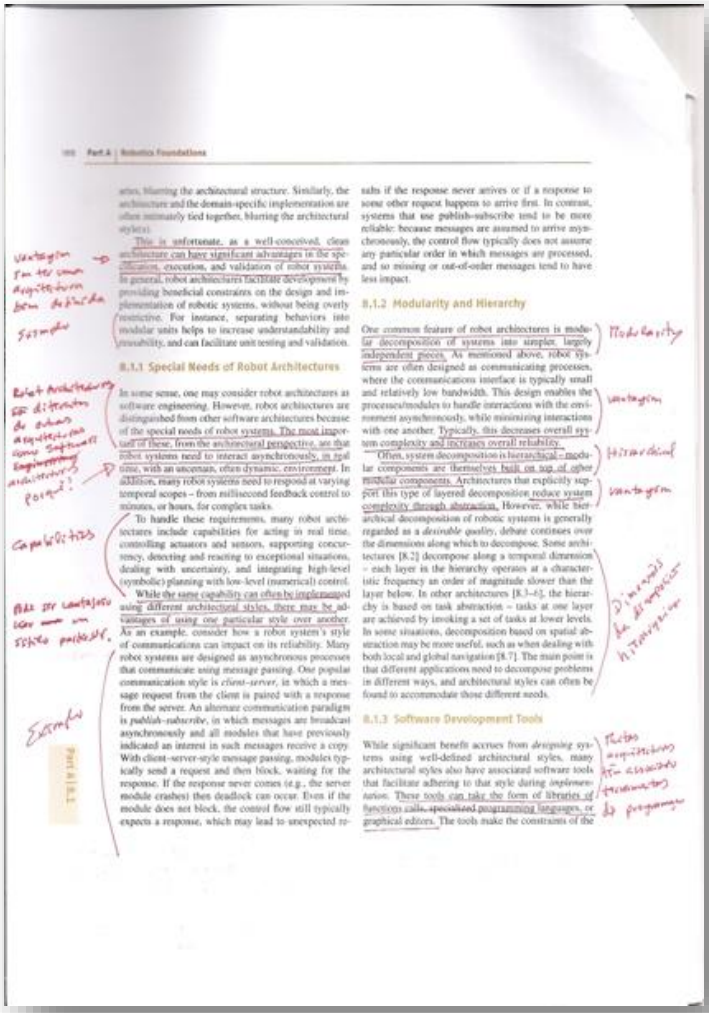
With the described technique the robot is able to track continuously its pose. However, the quality of its pose estimate will decrease according to the length of the traveled trajectory. We tackle this problem by actively distributing unique RFID tags into the environment, i.e. placing them automatically on the ground, and by utilizing the correspondences of these tags on the robots trajectory for calculating globally consistent maps according to the method introduced by Lu and Milios [8]. Suppose the robot distributes n RFID tags at unknown locations $l_1, l_2, ..., l_n$, with distance $d_{ij} = (\Delta x_{ij}, \Delta y_{ij}, \Delta\theta_{ij})$ between l_i and l_j. In order to determine the estimated distance $\hat{d}_{ij}$ with corresponding covariance matrix Σ_{ij} between l_i and l_j, the previously described Kalman filter is utilized. However, if the robot passes a tag l_i, we reset the Kalman Filter in order to estimate the *relative* distance $\hat{d}_{ij}$ to subsequent tag l_j on the robot's trajectory.

Our goal is now to estimate locations l_i that best explain the measured distances $\hat{d}_{ij}$ and covariances Σ_{ij}. This can be achieved with the maximum likelihood concept by minimizing the following Mahalanobis-distance:

$$W = \sum_{ij} \left(d_{ij} - \hat{d}_{ij}\right)^T \Sigma_{ij}^{-1} \left(d_{ij} - \hat{d}_{ij}\right), \quad (5)$$

where the summation goes over all measured distances and d_{ij} is the true distance between l_i and l_j. Note if we assume the robot's orientation to be measured by the IMU (whose error does not accumulate), we do not need to consider the orientation θ within the d_{ij} in Equation 5, and hence the optimization problem can be solved linearly by calculating

[2] From a series of experiments we determined $\Theta = 140$.



192 Part A | Robotics Foundations

aries, blurring the architectural structure. Similarly, the architecture and the domain-specific implementation are often intimately tied together, blurring the architectural style(s).

This is unfortunate, as a well-conceived, clean architecture can have significant advantages in the specification, execution, and validation of robot systems. In general, robot architectures facilitate development by providing beneficial constraints on the design and implementation of robotic systems, without being overly restrictive. For instance, separating behaviors into modular units helps to increase understandability and reusability, and can facilitate unit testing and validation.

8.1.1 Special Needs of Robot Architectures

In some sense, one may consider robot architectures as software engineering. However, robot architectures are distinguished from other software architectures because of the special needs of robot systems. The most important of these, from the architectural perspective, are that robot systems need to interact asynchronously, in real time, with an uncertain, often dynamic, environment. In addition, many robot systems need to respond at varying temporal scopes – from millisecond feedback control to minutes, or hours, for complex tasks.

To handle these requirements, many robot architectures include capabilities for acting in real time, controlling actuators and sensors, supporting concurrency, detecting and reacting to exceptional situations, dealing with uncertainty, and integrating high-level (symbolic) planning with low-level (numerical) control.

While the same capability can often be implemented using different architectural styles, there may be advantages of using one particular style over another. As an example, consider how a robot system's style of communication can impact on its reliability. Many robot systems are designed as asynchronous processes that communicate using message passing. One popular communication style is *client–server*, in which a message request from the client is paired with a response from the server. An alternate communication paradigm is *publish–subscribe*, in which messages are broadcast asynchronously and all modules that have previously indicated an interest in such messages receive a copy. With client–server-style message passing, modules typically send a request and then block, waiting for the response. If the response never comes (e.g., the server module crashes) then deadlock can occur. Even if the module does not block, the control flow still typically expects a response, which may lead to unexpected results if the response never arrives or if a response to some other request happens to arrive first. In contrast, systems that use publish–subscribe tend to be more reliable: because messages are assumed to arrive asynchronously, the control flow typically does not assume any particular order in which messages are processed, and so missing or out-of-order messages tend to have less impact.

8.1.2 Modularity and Hierarchy

One common feature of robot architectures is modular decomposition of systems into simpler, largely independent pieces. As mentioned above, robot systems are often designed as communicating processes, where the communications interface is typically small and relatively low bandwidth. This design enables the processes/modules to handle interactions with the environment asynchronously, while minimizing interactions with one another. Typically, this decreases overall system complexity and increases overall reliability.

Often, system decomposition is hierarchical – modular components are themselves built on top of other modular components. Architectures that explicitly support this type of layered decomposition reduce system complexity through abstraction. However, while hierarchical decomposition of robotic systems is generally regarded as a *desirable quality*, debate continues over the dimension along which to decompose. Some architectures [8.2] decompose along a temporal dimension – each layer in the hierarchy operates at a characteristic frequency an order of magnitude slower than the layer below. In other architectures [8.3–5], the hierarchy is based on task abstraction – tasks at one layer are achieved by invoking a set of tasks at lower levels. In some situations, decomposition based on spatial abstraction may be more useful, such as when dealing with both local and global navigation [8.7]. The main point is that different applications need to decompose problems in different ways, and architectural styles can often be found to accommodate those different needs.

8.1.3 Software Development Tools

While significant benefit accrues from *designing* systems using well-defined architectural styles, many architectural styles also have associated software tools that facilitate adhering to that style during *implementation*. These tools can take the form of libraries of functions calls, specialized programming languages, or graphical editors. The tools make the constraints of the

Anotações no próprio artigo

- Destaque frases importantes à medida que efetua a leitura.
  - Use cor em vez de simples sublinhado.
- Marque parágrafos importantes identificando o tipo de relevância dos mesmos:
  - Motivação/problema, ideias/solução, avaliação, contribuições.
- Escreva a mensagem a reter na primeira página do artigo.
- Escreva as questões levantadas, na primeira ou ultima página, ou nas margens e relacione-as com parágrafos ou frases específicas do artigo.

Dicas para encontrar artigos (pesquisa bibliográfica)

- Outros artigos
- Artigos sobre “estado da arte”
- Jornais científicos
- Atas de conferências científicas
- Sistemas de alerta das editoras
- Teses e Dissertações
- Grupos de Investigação e Investigadores
- Sugestões do supervisor e dos colegas

ATIVIDADE AVALIATIVA

- Aplicar a metodologia apresentada na leitura de um artigo da revista selecionada. (em duplas);
- Apresentação em slides indicando as respostas das questões apresentadas anteriormente: