

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ PROGRAMA DE  
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**MAINARA CRISTINA LORENCENA**

**UM *FRAMEWORK* PARA O CONTROLE E A SUPERVISÃO DE  
PROCESSOS AVÍCOLAS**

**DISSERTAÇÃO**

**PATO BRANCO  
2018**

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MAINARA CRISTINA LORENCENA

UM *FRAMEWORK* PARA O CONTROLE E A SUPERVISÃO DE  
PROCESSOS AVÍCOLAS

DISSERTAÇÃO

PATO BRANCO

2018

MAINARA CRISTINA LORENCENA

**UM *FRAMEWORK* PARA O CONTROLE E A SUPERVISÃO DE  
PROCESSOS AVÍCOLAS**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Elétrica da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de “Mestre em Engenharia Elétrica” – Área de Concentração: Sistemas e processamento de energia.

Orientador: Marcelo Teixeira

Co-orientador:Richardson Ribeiro

**PATO BRANCO**

**2018**

L868f

Lorencena, Mainara Cristina.

Um *framework* para o controle e supervisão de processos avícolas /  
Mainara Cristina Lorencena. -- 2018.

69 f. : il. ; 30 cm

Orientador: Prof. Dr. Marcelo Teixeira

Coorientador: Prof. Dr. Richardson Ribeiro

Dissertação (Mestrado) - Universidade Tecnológica Federal do Paraná.  
Programa de Pós-Graduação em Engenharia Elétrica. Pato Branco, PR,  
2018.

Bibliografia: f. 57 - 62.

1. Indústria avícola. 2. Sistemas de controle supervísório. 3. Modelagem.  
4. Automação. I. Teixeira, Marcelo, orient. II. Ribeiro, Richardson. III.  
Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação  
em Engenharia Elétrica. IV. Título.

CDD 22. ed. 621.3

Ficha Catalográfica elaborada por  
Suélem Belmudes Cardoso CRB9/1630  
Biblioteca da UTFPR Campus Pato Branco



---

## **TERMO DE APROVAÇÃO**

**Título da Dissertação n.º 060**

**“Um *Framework* para o Controle e a Supervisão de Processos Avícolas”**

por

**Mainara Cristina Lorencena**

Dissertação apresentada às oito horas e vinte minutos, do dia dezanove de março de dois mil e dezoito, como requisito parcial para obtenção do título de MESTRE EM ENGENHARIA ELÉTRICA, do Programa de Pós-Graduação em Engenharia Elétrica - Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco. A candidata foi arguida pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho APROVADO.

Banca examinadora:

---

**Prof. Dr. Marcelo Teixeira**  
UTFPR/PB (Orientador)

---

**Prof. Dr. Fábio Luiz Bertotti**  
UTFPR/PB

---

**Prof. Dr. Dalcimar Casanova**  
UTFPR/PB

---

**Prof. Dr. Willian Zakewski**  
UNILA/Foz do Iguaçu

---

**Prof. Dr. Jean Patric da Costa**  
Coordenador do Programa de Pós-Graduação em  
Engenharia Elétrica - PPGEE/UTFPR

A via original devidamente assinada, encontra-se na Biblioteca da UTFPR – Câmpus Pato Branco.

Dedico este trabalho aos meus pais, Neide Lorencena e Idelmar Lorencena, e à minha irmã Andrieli Lorencena.

## AGRADECIMENTOS

Agradeço aos meus pais, Neide Lorencena e Idelmar Lorencena por todo apoio e incentivo dado desde a infância, e à minha irmã, Andrieli Lorencena, pelo amor e companheirismo de sempre. Sou grata a Deus por ter me presenteado com uma família tão especial e por ter me permitido viver esta conquista.

Ao meu orientador, Marcelo Teixeira, por ter exercido seu trabalho com o máximo de dedicação, auxiliando-me em todos os momentos, e aos demais professores que muito contribuíram com minha formação profissional.

Aos amigos que, perto ou distantes, torceram por mim. Vocês fazem parte da minha história. Obrigada!

Tente mover o mundo - o primeiro passo será mover a si mesmo.

Platão.



## RESUMO

LORENCENA, Mainara. UM *FRAMEWORK* PARA O CONTROLE E A SUPERVISÃO DE PROCESSOS AVÍCOLAS. 69 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Pato Branco, 2018.

O manejo avícola é uma atividade criteriosa a ser realizada em galpões de criação de frangos de corte. Nele estão inseridas tarefas como manutenção da higiene, sanidade e, principalmente, controle de ambiência. A temperatura e a umidade são os principais parâmetros para que se atinja o equilíbrio térmico. No entanto, grande parte dos sistemas de controle de ambiência para aviários disponíveis no mercado não apresentam a eficiência desejável, caracterizada pela manutenção da temperatura e da umidade e máxima homogeneização possível destas variáveis ao longo do ambiente. Ao contrário disso, o que se observa é que, mesmo na presença de sistemas semiautomáticos de controle, as principais decisões de manejo ainda dependem da percepção do especialista (avicultor) para observar o processo e configurar os parâmetros do controlador. Além de ser ineficiente, esta atividade também é incômoda e propensa a erros. Esta dissertação propõe um *framework* para o controle, a supervisão e a tomada de decisões em processos avícolas. Inicialmente, é sistematizada a coleta de um conjunto de eventos relacionados à temperatura e umidade, de maneira a prover mais detalhes sobre as condições climáticas internas do aviário. A partir disso, são apresentadas as etapas de modelagem e de síntese de controle com base na *Teoria de Controle Supervisório (TCS)*, em suas versões *monolítica* e *modular local*, as quais são comparadas e fundamentam a implementação física, em microcontrolador. As vantagens da abordagem escolhida em relação à homogeneização da temperatura e da umidade em aviários são ilustradas via simulação e comparadas a um cenário tradicional de manejo. Resultados finais ainda fornecem uma opção de supervisão remota do processo utilizando uma ferramenta SCADA.

**Palavras-chave:** Automação, Controle, Modelagem Formal, Processo Avícola

## ABSTRACT

LORENCENA, Mainara. A FRAMEWORK FOR CONTROL AND SUPERVISION OF POULTRY FARMING. 69 f. Dissertação – Programa de Pós-graduação em Engenharia Elétrica, Universidade Tecnológica Federal do Paraná. Pato Branco, 2018.

Poultry management is a judicious activity to be carried out in poultry farms. In it are inserted tasks like maintenance of hygiene, sanity and, mainly, ambience control. Temperature and humidity are the main parameters for achieving thermal equilibrium. However, most ambience control systems for aviaries available on the market don't have the desirable efficiency, characterized by the maintenance of temperature and humidity and maximum possible homogenization of these variables throughout the environment. Instead, what is observed is that, even in the presence of semiautomatic control systems, the main management decisions still depend on the expert's perception (aviculturist) to observe the process and configure the parameters of the controller. In addition to being inefficient, this activity is also cumbersome and prone to errors. This dissertation proposes a framework for control, supervision and decision making in poultry process. Initially, it is systematized the collection of a set of events related to temperature and humidity, in order to provide more details about the internal climatic conditions of the aviary. From this, the modeling and control synthesis steps are presented based on the Supervisory Control Theory (TCS), in its monolithic and modular versions, which are compared, and give support the physical implementation in microcontroller. The advantages of the approach chosen in relation to the homogenization of temperature and humidity in aviaries are illustrated by simulation and compared to a traditional management scenario. Final results still provide a remote process supervision option using a SCADA tool.

**Keywords:** Automation, Control, Formal Modeling, Poultry Farming

## LISTA DE FIGURAS

FIGURA 1	– Exemplo de composição síncrona .....	16
FIGURA 2	– Arquitetura de controle CMC. ....	20
FIGURA 3	– Arquitetura de controle CML. ....	21
FIGURA 4	– Formação de supervisores locais no CML. ....	22
FIGURA 5	– Arquitetura com fluxo de ar longitudinal .....	25
FIGURA 6	– Arquitetura com fluxo de ar transversal. ....	28
FIGURA 7	– Método que conduz a uma ferramenta de monitoramento para um processo controlado de avicultura .....	31
FIGURA 8	– Modelos de atuadores .....	32
FIGURA 9	– Sensor models .....	34
FIGURA 10	– Exemplo de modelagem de especificação .....	38
FIGURA 11	– Modelos de especificação para a operação do exaustor .....	39
FIGURA 12	– Modelos de especificação para o funcionamento do <i>pad-cooling</i> ..	40
FIGURA 13	– Modelos de especificação para o funcionamento da cortina .....	42
FIGURA 14	– Modelo de especificação para acionamento e desligamento do aquecedor .....	43
FIGURA 15	– Simulação de uma ação proibitiva no software Supremica por meio de especificações .....	43
FIGURA 16	– Exemplo de construção de supervisores locais na ferramenta Supremica .....	48
FIGURA 17	– Exemplo de estruturas modulares .....	49
FIGURA 18	– Interface gráfica utilizando componentes do software ScadaBR ..	53
FIGURA 19	– Modelo tradicional de aviário para simulação no <i>SolidWorks</i> .....	64
FIGURA 20	– Modelo proposto de aviário para simulação no <i>SolidWorks</i> .....	64
FIGURA 21	– Distribuição das faixas internas de temperatura via simulação na arquitetura com fluxo de ar longitudinal (quadro superior) e na arquitetura com fluxo de ar transversal (quadro inferior) .....	65
FIGURA 22	– Distribuição das faixas internas de umidade via simulação na arquitetura com fluxo de ar longitudinal (quadro superior) e na arquitetura com fluxo de ar transversal (quadro inferior) .....	66
FIGURA 23	– Ciclo de mensagens genéricas do Modbus trocadas entre mestre e escravo .....	68
FIGURA 24	– Composição do pacote de informações do Modbus RTU .....	68

## LISTA DE TABELAS

TABELA 1 – Estatística da composição dos supervisores locais. ....	49
--	----

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>14</b>
2.1	SISTEMAS A EVENTOS DISCRETOS	14
2.2	MODELAGEM DE SEDS	15
2.3	TEORIA DE CONTROLE SUPERVISÓRIO	17
2.3.1	Complexidade da síntese monolítica	19
2.4	ABORDAGEM MODULAR	20
2.4.1	Complexidade de síntese no CMC	20
2.4.2	Controle Modular Local	21
2.4.2.1	Complexidade da síntese Modular Local	23
2.5	PROCESSO AVÍCOLA	24
2.5.1	Arquitetura com fluxo de ar longitudinal	24
2.6	ARQUITETURA COM FLUXO DE AR TRANSVERSAL	27
<b>3</b>	<b>MÉTODO</b>	<b>30</b>
3.1	MÉTODO DE MODELAGEM, CONTROLE E SUPERVISÃO	30
3.2	MODELAGEM DA PLANTA DO SISTEMA POR AUTÔMATOS FINITOS	31
3.2.1	Modelagem de Atuadores	31
3.2.1.1	Modelo do Exaustor	32
3.2.1.2	Modelo do <i>Pad-cooling</i>	33
3.2.1.3	Modelo da Cortina	33
3.2.1.4	Modelo do Aquecedor	33
3.2.2	Modelagem dos sensores	34
3.2.2.1	Modelo do sensor de temperatura	34
3.2.2.2	Modelo do sensor de umidade	35
3.2.3	Composição da planta	36
3.3	MODELAGEM DAS ESPECIFICAÇÕES DO SISTEMA POR AUTÔMATOS FINITOS	36
3.3.1	<i>Design</i> sistemático	37
3.3.1.1	Coordenação do exaustor	39
3.3.1.2	Coordenação do nebulizador	40
3.3.1.3	Coordenação da cortina	41
3.3.1.4	Aquecedor	42
3.3.2	Especificação global	43
<b>4</b>	<b>SÍNTESE DE CONTROLE: ANÁLISE ESTATÍSTICA</b>	<b>45</b>
4.1	ESTATÍSTICA DE CONTROLE MONOLÍTICO	45
4.2	ESTATÍSTICA DE CONTROLE MODULAR LOCAL	47
<b>5</b>	<b>ASPECTOS DE IMPLEMENTAÇÃO</b>	<b>50</b>
5.1	GERAÇÃO DE CÓDIGO EM LINGUAGEM C	50
5.2	MONITORAMENTO DO SISTEMA SOB CONTROLE	52
<b>6</b>	<b>CONCLUSÃO</b>	<b>54</b>
	REFERÊNCIAS	57

<b>Apêndice A - SIMULAÇÃO DOS PARÂMETROS DE AMBIÊNCIA .....</b>	<b>63</b>
<b>Apêndice B - PROTOCOLO MODBUS RTU .....</b>	<b>67</b>

## 1 INTRODUÇÃO

Com papel fundamental na economia brasileira e no desenvolvimento regional, a avicultura emprega hoje mais de 3,6 milhões de pessoas, responde por quase 1,5% do *Produto Interno Bruto* (PIB) brasileiro (ABPA, 2015; UBABEF, 2008) e projeta o país como um dos maiores produtores mundiais nessa atividade, sendo o Paraná o estado que mais produz frangos no Brasil (CIAS, 2016).

O manejo avícola é um dos procedimentos mais importantes na criação de frangos de corte e a principal diretriz do manejo é o conforto térmico (CANGAR et al., 2008; AMARAL et al., 2011). Proporcionar conforto térmico, por sua vez, associa-se ao controle adequado de 2 fatores principais: (i) temperatura e (ii) umidade relativa do ar (PRINCE et al., 1965; REECE et al., 1972; DONKOH, 1989; DAWKINS et al., 2004). Esses dois fatores interferem no ganho de peso das aves ao longo de seu curto ciclo de vida, em contrapartida à quantidade de alimento que consomem, relação a qual é denominada *conversão alimentar* (NOBLE; TEETER, 2004; NASCIMENTO et al., 2005). Especialistas definem a chamada *zona de conforto* (CANDIDO et al., 2016; NASCIMENTO et al., 2011) para quantificar os níveis de temperatura e de umidade desejadas para as aves em cada estágio de vida (HULZEBOSCH, 2006; BRUZUAL et al., 2000; AVIAGEN, 2014; VANTRESS, 2008).

Fora de sua zona de conforto, a ave pode sofrer de estresse térmico, tendo seu desenvolvimento prejudicado tanto por altas quanto por baixas temperaturas e por níveis baixos ou elevados de umidade relativa do ar (PEREIRA; NÄÄS, 2008; XIN et al., 1994; SOUZA et al., 2010). A alta temperatura ambiente influencia diretamente na homeotermia, relacionada à função fisiológica de eliminar o excesso de calor corporal (DONKOH; ATUAHENE, 1988). Por outro lado, quando a ave é submetida a temperaturas abaixo da região de conforto, ela destina parte da energia ingerida para gerar calor corporal, o que reduz a produtividade (DOWELL, 1974). Por essas razões, a preocupação com o conforto térmico é determinante na atividade avícola, cedendo espaço para novas técnicas de automatização (GATES; XIN, 2008; MORTENSEN et al.,

2016; SO-IN et al., 2014; AYDIN, 2017).

No entanto, o que se vê na avicultura destoa das necessidades da indústria por melhorias e produtividade. O controle térmico, como é feito atualmente por meio dos sistemas providos pelas empresas que centralizam a produção (chamadas *integradoras*), é um procedimento dependente sobretudo da percepção do avicultor para o ajuste dos parâmetros de temperatura e de umidade na interface gráfica do sistema de controle. Isso, além de requerer a sua presença constante no aviário, remete a uma prática propensa a erros, uma vez que em um aviário extenso (100m a 200m de comprimento) e heterogêneo, como é o caso, pode não ser trivial identificar os setores específicos que demandam ação de controle térmico.

Essa dificuldade é, em grande parte, associada à disposição inadequada dos equipamentos ao longo do aviário, o que, conseqüentemente, remete a um sistema de controle ineficaz (BRIGANO, 2013). Normalmente, de 1 a 2 sensores são posicionados no ambiente para a leitura de dados. Então, atuadores, que geralmente se localizam nos extremos da estrutura, são acionados automaticamente conforme a leitura desses sensores. Esse cenário dificulta a preservação de um ambiente homogêneo para as aves, pois diferentes setores podem apresentar condições climáticas distintas, em razão da extensão da estrutura, da incidência de sol/sombra, etc. Na prática, os atuadores são acionados manualmente em função do que o avicultor observa nas aves, já que elas apresentam comportamentos distintos para calor e frio (PEREIRA et al., 2013).

Para um controle de ambiência mais eficiente, é ideal que os galpões contassem com total isolamento interno, impedindo a influência de condições climáticas externas. Em contrapartida, é fundamental a circulação e renovação mínima de ar (CRUZ, 2003), pois a falta de ventilação pode ocasionar o aumento da umidade relativa do ar e a concentração de gases tóxicos, como a amônia, o dióxido de carbono, etc. (RONCHI, 2004). Esses argumentos remetem à escolha de uma abordagem de automação que absorva as necessidades específicas do manejo avícola em ambiente controlado, equilibrando o conforto térmico e, conseqüentemente, potencializando a produtividade (MENEGALI, 2005).

Na prática da automação industrial, incluindo a de processos avícolas, uma etapa fundamental é a programação das sequências operacionais (obtenção da lógica de controle) responsável pela forma como o sistema é comandado sob controle. Essa programação é, em geral, realizada manualmente, o que é passível de erros e dependente da experiência do programador. Nesse sentido, abordagens baseadas em mo-



delos fornecem mecanismos formais capazes de descrever a síntese de controladores ótimos, ou seja, controladores que possuem a propriedade de atuar sobre o modelo de um sistema de forma *minimamente restritiva*, não-bloqueante e em consonância com um conjunto de especificações. A *Teoria de Controle Supervisório* (TCS) (RAMADGE; WOHNHAM, 1989) de *Sistema e Eventos Discretos* (SEDs) (CASSANDRAS; LAFORTUNE, 2008), por exemplo, é uma abordagem formal que permite computar controladores para uma classe particular de sistemas que não dependem do tempo, mas de eventos assíncronos no tempo.

Este trabalho utiliza a TCS para sistematizar a obtenção de uma lógica de controle automático que promova o equilíbrio térmico no interior de aviários. Inicialmente, é definida uma planta física do aviário de maneira que os equipamentos presentes sejam capazes de prover mais detalhes sobre as condições climáticas internas, sistematizando a coleta de um conjunto de eventos relacionados à temperatura e à umidade em diferentes setores. Contudo, dispor de mais eventos implica naturalmente em uma maior complexidade de controle pois certas ações passam a depender não só de um evento, mas da orquestração de um conjunto deles. Nesse sentido, esta dissertação também apresenta as etapas de modelagem, síntese, implementação e supervisão de processos avícolas. Na fase de design são obtidos modelos que representam o comportamento de cada dispositivo disposto ao longo da extensão do aviário. Esses modelos são então manipulados por operações matemáticas (conforme a TCS) que sintetizam um controlador que contém tão somente as sequências operacionais corretas para que o sistema evolua sob controle. Duas versões da TCS, monolítica (RAMADGE; WOHNHAM, 1989) e modular (QUEIROZ; CURY, 2000), são utilizadas e comparadas, justificando as escolhas na fase de implementação.

O controlador obtido é implementado em hardware e uma interface gráfica SCADA (*Supervisory Control and Data Acquisition*) é provida para auxiliar no manejo avícola por meio do monitoramento e do controle remoto (LORENCENA, 2016). O software SCADA utilizado é o ScadaBR. O controlador se comunica com a interface gráfica por meio do protocolo de comunicação *Modbus RTU*. Dessa forma, o especialista pode visualizar em tempo real as condições do interior do aviário e, se for o caso, interferir nas decisões do controlador automático, acionando ou desligando certos equipamentos, sobrepondo a decisão do especialista às ações de controle, sem necessariamente estar presente no aviário.

## 2 REFERENCIAL TEÓRICO

Este capítulo aborda os conceitos relacionados a TCS a serem empregados na obtenção de um controlador para processos avícolas.

Na sequência é relatado o problema de controle de ambiência em processos avícolas apresentando duas arquiteturas diferentes para galpões de frangos de corte e a disposição de sensores e atuadores em cada uma.

Neste contexto, o controle de ambiência em aviários é visto como um SED.

### 2.1 SISTEMAS A EVENTOS DISCRETOS

Grande parte dos processos industriais, incluindo os processos avícolas, possuem uma dinâmica dirigida por eventos, o que define uma classe de processos denominados *Sistemas a Eventos Discretos* (SEDs) (CASSANDRAS; LAFORTUNE, 2008). Os SEDs compartilham de um comportamento definido por um conjunto enumerável de estados tal que a evolução entre esses estados depende de sinais esporádicos, assíncronos e independentes no tempo, denominados *eventos*.

Entre a ocorrência de dois eventos consecutivos, um SED permanece num determinado estado. A ocorrência de um evento causa então uma transição de estado, de forma que sua evolução no tempo pode ser representada pela trajetória percorrida unicamente no seu espaço de estados (CURY, 2001). Exemplos de SEDs incluem sistemas robóticos, de manufatura, de supervisão de tráfego, de logística, gerenciadores de bases de dados, processos agrícolas e avícolas, entre outros.

Os SEDs podem ser representados por estruturas computacionais que permitem a simulação e a implementação destes sistemas. Para isso, é necessário um modelo que descreva o comportamento global e o comportamento desejado para cada caso. A seguir, é apresentada uma abordagem para representação e modelagem de SEDs.

## 2.2 MODELAGEM DE SEDS

Uma estrutura computacional que pode ser usada para descrever o comportamento de SEDs são as *linguagens formais* (CASSANDRAS; LAFORTUNE, 2008). *Eventos* definem as estruturas básicas de uma linguagem e são elementos tomados de um *alfabeto* finito  $\Sigma$ , em que  $\Sigma^*$  denota o conjunto de todas as *cadeias* finitas de eventos construídas em  $\Sigma$ , incluindo a *cadeia vazia*  $\varepsilon$ . Um subconjunto  $L \subseteq \Sigma^*$  é chamado de *linguagem*. Uma linguagem  $L$  definida sobre um alfabeto  $\Sigma^*$  é um conjunto de cadeias formadas de eventos em  $\Sigma^*$ . Uma linguagem  $L$  é dita ser *regular* se composta somente por classes equivalentes finitas (SHALLIT; SHALLIT, 2009).

Na indústria, a propriedade de regularidade é de interesse pois, nesse caso, uma linguagem se torna adequada para tratamento computacional e pode ser representada por estruturas finitas de modelos, como por exemplo os *Autômatos Finitos* (AFs).

Formalmente, um AF é uma 5-tupla  $A = \langle \Sigma, Q, q^\circ, Q^\omega, \rightarrow \rangle$ , em que:

- $\Sigma$  é o alfabeto de eventos;
- $Q$  é o conjunto de estados;
- $q^\circ \in Q$  é o estado inicial;
- $Q^\omega \subseteq Q$  é o subconjunto de estados marcados, associados à ideia de tarefas completas; e
- $\rightarrow \subseteq Q \times \Sigma \times Q$  é a relação de transição.

Para dois estados  $q_1, q_2 \in Q$ , denota-se por  $q_1 \xrightarrow{\sigma} q_2$  uma transição do estado  $q_1$  para o estado  $q_2$  com o evento  $\sigma \in \Sigma$ .  $A \xrightarrow{s} q$  denota que uma cadeia  $s$  é possível em  $A$ .

Duas linguagens podem ser definidas a partir de  $A$ :

$$L(A) = \{s \in \Sigma^* \mid A \xrightarrow{s} q \in Q\} \text{ e}$$

$$L^\omega(A) = \{s \in \Sigma^* \mid A \xrightarrow{s} q \in Q^\omega\}.$$

$L(A)$  é a *linguagem gerada*, contendo todas as cadeias reconhecidas por  $A$ , enquanto  $L^\omega(A)$  é a *linguagem marcada*, i.e., o conjunto de todas as cadeias que levam a estados marcados.

Dois AFs  $A = \langle \Sigma_A, Q_A, q_A^\circ, Q_A^\omega, \rightarrow_A \rangle$  e  $B = \langle \Sigma_B, Q_B, q_B^\circ, Q_B^\omega, \rightarrow_B \rangle$  podem ser associados por composição síncrona, operação matemática denotada por  $\parallel$ , que é definida tal que:

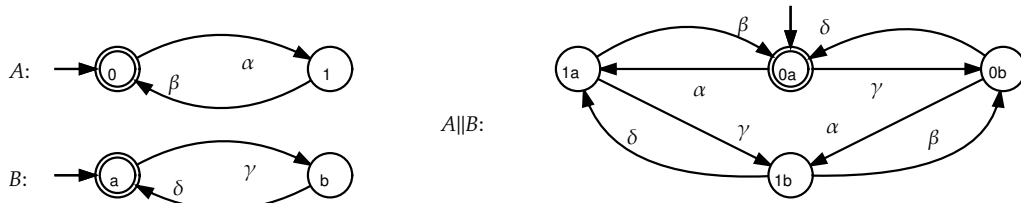
$$A \parallel B = \langle \Sigma_A \cup \Sigma_B, Q_A \times Q_B, (q_A^\circ, q_B^\circ), Q_A^\omega \times Q_B^\omega, \rightarrow \rangle,$$

em que:

- $(q_A, q_B) \xrightarrow{\sigma} (q'_A, q'_B)$ , se  $\sigma \in \Sigma_A \cap \Sigma_B$ ;
- $(q_A, q_B) \xrightarrow{\sigma} (q'_A, q_B)$ , se  $\sigma \in \Sigma_A \setminus \Sigma_B$ ;
- $(q_A, q_B) \xrightarrow{\sigma} (q_A, q'_B)$ , se  $\sigma \in \Sigma_B \setminus \Sigma_A$ .

Na composição síncrona, os eventos compartilhados pelos autômatos envolvidos na operação são sincronizados, i.e., as transições são fundidas para o mesmo evento. Do contrário, ou seja, caso os eventos não sejam compartilhados pelos AFs, então eles sofrem o *interleaving*, ou seja, as transições não são fundidas e o modelo evolui em qualquer ordem.

A Figura 1 ilustra a composição de dois AFs  $A$  e  $B$ .



**Figura 1:** Exemplo de composição síncrona

Para um conjunto de AFs  $P^i = \langle \Sigma_i, Q_i, q_i^\circ, Q_i^\omega, \rightarrow_i \rangle$ , para  $i = 1, \dots, n$ , um modelo global  $P = \langle \Sigma_P, Q_P, q_P^\circ, Q_P^\omega, \rightarrow_P \rangle$  pode ser obtido compondo-se  $P = \parallel_{i=1}^n P^i$ , tal que  $\Sigma_P = \bigcup_{i=1}^n \Sigma_i$ . Composições são particularmente úteis por permitirem que um sistema seja expressado modularmente, e, em tese, de modo mais simples.

Um estado  $q \in Q_P$  é dito ser *acessível* se  $\exists s \in \Sigma_P^*$ , tal que  $q_0 \xrightarrow{s} q$ . Assim,  $P$  é acessível se  $q$  é acessível para todo  $q \in Q_P$ . Um *componente acessível* de  $P$ ,  $P_{ac}$ , pode ser obtido simplesmente removendo-se os estados não-acessíveis de  $P$ , bem como as suas transições associadas. Um estado não-acessível é dito um “mau-estado” (CURY et al., 2015) (MOOR, 2016).

$P$  é dito ser *co-acessível* se cada cadeia  $s \in L(P)$  é um prefixo de uma cadeia marcada. Na prática isso representa que cada  $s \in L(P)$  pode ser completada por alguma  $s' \in \Sigma_P^*$  tal que  $ss' \in L^\omega(P)$ , i.e.,  $L(P) = \overline{L^\omega(P)}$ . Então, um *componente co-acessível* de  $P$ ,  $P_{co}$  pode ser obtido simplesmente removendo-se os estados (e transições associadas) que são não co-acessíveis em  $P$ , i.e., estados alcançados por prefixos de cadeias incapazes de alcançar um estado marcado.

Se  $P$  é co-acessível, ele é dito ser um AF *não-bloqueante*. Portanto, quando  $L(P) = \overline{L^\omega(P)}$ , o comportamento modelado por  $P$  é não-bloqueante. Se  $L(P) \neq \overline{L^\omega(P)}$  então existe  $z \in L(P)$  tal que  $z \notin \overline{L^\omega(P)}$ . Se  $P$  for ao mesmo tempo acessível e co-acessível, ele é dito ser *trim*. Um *componente trim* de  $P$ ,  $P_{trim}$ , pode ser obtido pela fusão de  $P_{ac}$  e  $P_{co}$ .

### 2.3 TEORIA DE CONTROLE SUPERVISÓRIO

Quando os componentes de um sistema são modelados declarativamente, como por meio de AFs, a composição desses modelos individuais leva a uma estrutura global que expressa a planta do sistema em *malha aberta*, ou seja, livre de restrições.

Sob o ponto de vista prático, uma planta precisa ser ajustada a um determinado comportamento para quando estiver em funcionamento. Esse ajuste é implementado por uma segunda estrutura de modelos denominada *especificação*. Especificação é considerada uma ação proibitiva que, ao ser associada ao modelo da planta (via composição síncrona), *controla* o seu comportamento conforme os requisitos especificados, o que gera um comportamento em *malha fechada*. Um sistema pode possuir uma especificação ou um conjunto de especificações, e cada uma também pode ser representada individualmente por AFs, compostos a posteriori.

Sejam dois AFs  $G = \parallel_{j=1}^m G^j$  e  $E = \parallel_{i=1}^n E^i$ , modelando respectivamente a planta e as especificações de um SED. A composição  $K = G \parallel E$ , tal que  $L^\omega(K) \subseteq L^\omega(G)$ , modela o comportamento do sistema em malha fechada. Nesse caso,  $K$  também reflete o comportamento desejado, ou seja, a materialização das ações de controle exatamente da forma como projetadas pelo engenheiro. Essa estrutura pode ser utilizada para fins de implementação. Entretanto, usar  $K$  para implementar um controlador significa assumir que qualquer evento cuja ocorrência seja possível na planta, é passível de intervenção (controle). Na prática, essa abordagem pode não ser consistente, haja visto que alguns eventos possuem uma característica tal que não permitem ser diretamente controlados. É o caso, por exemplo, de um evento que representa o recebimento de

um pacote de dados em um sistema de comunicação. O evento pode ser habilitado na planta, mas não se tem domínio sobre sua ocorrência. Outros eventos não controláveis incluem finais de operações, paradas de equipamentos, etc. Se tais eventos não podem ser diretamente evitados, então é intuitivo pensar que a estrutura que modela esses eventos deve incorporar esse aspecto, por questões de consistência entre o sistema e as respectivas ações de controle.

Uma abordagem formal que permite incorporar a controlabilidade de eventos na síntese de controladores para SEDs é a *Teoria de Controle Supervisório* (TCS) (RAMADGE; WONHAM, 1989). Na TCS, o conjunto de eventos é particionado tal que  $\Sigma = \Sigma_c \cup \Sigma_u$ , em que  $\Sigma_c$  contém os *eventos controláveis*, cuja ocorrência pode ser evitada na planta, e  $\Sigma_u$  contém os *eventos não-controláveis*, os quais não são diretamente manipuláveis.

O elemento da TCS que efetivamente implementa as ações de controle na planta, sintetizando a questão da controlabilidade, é o *supervisor*. Um supervisor  $S$  é uma estrutura associada à planta  $G$  que, após qualquer cadeia  $s \in L(G)$ , observa os eventos possíveis na planta e informa, dentre eles, quais devem ser habilitados.

A ação de controle de  $S$  sobre  $G$  ( $S/G$ ) consiste em habilitar eventos do conjunto  $S(s) \subseteq \Sigma$ , com  $\Sigma_u \subseteq S(s)$ . Eventos possíveis na planta que não pertençam a  $S(s)$  são inibidos pela ação de controle. O *comportamento gerado em malha fechada* é denotado por  $L(S/G)$  e corresponde ao conjunto de cadeias que sobrevivem sob controle. Já o *comportamento marcado em malha fechada* é dado por  $L^\omega(S/G) = L(S/G) \cap L_S$ , em que  $L_S \subseteq L^\omega(G)$ .  $S$  é não bloqueante quando  $L(S/G) = \overline{L^\omega(S/G)}$ .

O *Problema de Controle Supervisório* (PCS) consiste, então, em obter um supervisor  $S$  tal que sua ação sobre  $G$  ( $S/G$ ) satisfaça o conjunto de especificações  $K$  ( $L^\omega(S/G) \subseteq K$ ) e, para isso, desabilite apenas eventos controláveis.

A *controlabilidade* é uma condição necessária e suficiente para a existência de  $S$ . Uma linguagem  $K \subseteq \Sigma^*$  é *controlável* em relação a uma linguagem prefixo-fechada  $L(G)$  se

$$\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$$

Ou seja, se após qualquer prefixo  $s$ , um evento  $\mu \in \Sigma_u$  é elegível em  $L(G)$ , então  $\mu$  não é desabilitado, i.e.,  $s\mu \in \overline{K}$ .

Se uma especificação  $K \subseteq L(G)$  é controlável em relação a  $L(G)$ , então a solução de controle pode ser implementada por um autômato  $V$  que representa  $K$ , para  $K =$

$L^\omega(V) \cap L^\omega(G)$  e  $\bar{K} = L(V) \cap L(G)$ . Se  $K$  não for controlável, se faz necessário reduzi-la à sua *máxima sublinguagem controlável*

$$\text{supC}(K, G) = \bigcup \{ K' \subseteq K \mid K' \text{ é controlável em relação a } L(G) \}. \quad (1)$$

O processo de computar  $\text{supC}(K, G)$  é conhecido como *síntese* (CASSANDRAS; LAFORTUNE, 2008) e nada mais é do que uma operação matemática que implementa a noção de controlabilidade, i.e., o algoritmo de síntese extrai de  $K$  um sub-modelo  $K'$  que trata da impossibilidade de interferir diretamente em eventos em  $\Sigma_u$ .

Assim,  $\text{supC}(K, G)$  representa o comportamento menos restritivo possível de ser implementado por um supervisor  $S$  ao controlar  $G$ , respeitando a especificação  $K$ . Se  $\text{supC}(K, G)$  for ainda não-bloqueante, então  $L^\omega(S/G) = \text{supC}(K, G)$  e  $L^\omega(S/G) = \overline{\text{supC}(K, G)}$  é uma solução *ótima* para o PCS.

### 2.3.1 COMPLEXIDADE DA SÍNTESE MONOLÍTICA

A complexidade computacional, envolvida em uma operação monolítica de síntese, é polinomial no número de estados de  $G$  e  $E$  (RAMADGE; WONHAM, 1987b). Por conseguinte, tais operações são consideradas computacionalmente tratáveis, já que uma resposta pode ser obtida em um tempo que é limitado por uma função polinomial.

Na prática, porém, o grande empecilho computacional da TCS reside no fato de que a estrutura  $G||E$  pode apresentar um aumento do número de estados de ordem exponencial. Por exemplo, seja um SED modelado por  $G$  e restrito por  $E$ , compostos como segue.

$$G = \prod_{j=1}^p G^j \quad E = \prod_{i=1}^e E^i$$

Considerando-se que cada  $G^j$  tem no máximo  $m$  estados, então  $G$  possui  $m^p$  estados, no pior caso. Ainda, se cada  $E^i$  possui no máximo  $n$  estados, então  $E$  pode possuir até  $n^e$  estados. Logo, a operação monolítica de síntese possui complexidade na ordem de  $O[(n^e \cdot m^p)^2]$ .

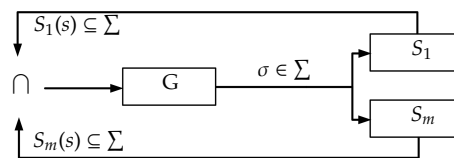
Portanto, o cálculo de um supervisor monolítico, embora seja polinomial no número de estados de  $G || E$ , é exponencial na quantidade de modelos envolvidos no problema. Isso significa que, embora uma operação de síntese seja computável, ela pode não ser tratável na prática. A solução para muitos problemas de controle é inviabilizada por esse entrave computacional, principalmente para aqueles problemas

que requerem uma solução centralizada, mas que envolvem múltiplos subsistemas e especificações, o que é típico em SEDs de grande porte.

## 2.4 ABORDAGEM MODULAR

Uma alternativa que tem surgido na literatura para lidar com o controle de SEDs de grande porte é a modularização do problema, de modo que partes possam ser tratadas individualmente e, em tese, de maneira mais simples.

No Controle Modular Clássico (CMC) (RAMADGE; WONHAM, 1987a), ao invés de se construir um único supervisor  $S$  que atenda a um conjunto global de especificações  $I = \{1, \dots, m\}$ , sintetiza-se um supervisor modular  $S_i$  para cada especificação  $i \in I$ . A estrutura de controle do CMC é apresentada na Figura 2.



**Figura 2:** Arquitetura de controle CMC.

Toda vez que, após uma cadeia  $s \in \Sigma^*$  qualquer, um evento  $\sigma \in \Sigma$  for observado na planta, cada supervisor modular mapeia um subconjunto  $S_i(s) \subseteq \Sigma$  de eventos a serem habilitados. Assim, se um evento é observado e não fizer parte de ambos os subconjuntos  $S_i(s)$ , ele é desabilitado na planta.

Um dos principais benefícios do CMC é a flexibilidade agregada pela resolução de um problema de controle por módulos, o que permite alterar, inserir ou remover um requisito em particular, sem afetar o sistema como um todo. Além disso, as operações individuais de síntese são, em geral, mais simples se comparadas à uma operação monolítica.

### 2.4.1 COMPLEXIDADE DE SÍNTESE NO CMC

Seja um SED modelado por  $p + q$  subsistemas, com no máximo  $m$  estados cada, de modo que  $m^{p+q}$  é o número de estados do autômato que modela a planta  $G$ . Ainda, seja  $G$  restrito pelas especificações  $E^i$ ,  $i \in I = \{1, \dots, e\}$ , tendo  $E^i$  no máximo  $n$  estados. Considerando que cada  $E^i$  restringe apenas uma parte  $p$  dos subsistemas  $p + q$ , e que  $q$  compõe os subsistemas que não são afetados por nenhuma das especificações, então  $q$  não precisa fazer parte da síntese dos supervisores modulares.

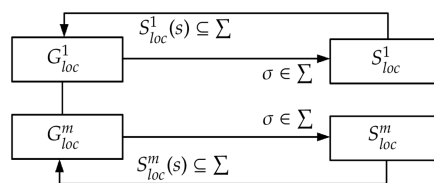


Logo, a complexidade de síntese no CMC é na ordem de  $O[(n.m^p)^2]$  e, portanto, é claramente mais simples do que no PCS centralizado, em que a ordem é  $O[(n^e.m^{p+q})^2]$ . Na prática, as vantagens do CMC em relação ao PCS vertem de duas fontes principais: (i) do fato de que as especificações não são compostas entre si; e (ii) do uso de um modelo parcial da planta na síntese, o qual não contempla os subsistemas assíncronos com as especificações.

Porém, note que é pouco razoável que um subsistema que não precise ser controlado (não compartilhe eventos com nenhuma especificação) faça parte do problema de controle. Logo, pode-se assumir que a planta  $G$  compõe, na verdade, todos os  $p + q$  subsistemas. Como o fator  $p + q$  determina o crescimento de  $G$ , então o CMC se torna degradado pela impossibilidade de tratar de problemas de grande porte. Uma extensão do CMC, o *Controle Modular Local*, revitaliza o conceito de controle supervisorio modular, e é apresentada a seguir.

#### 2.4.2 CONTROLE MODULAR LOCAL

O *Controle Modular Local* (CML) (QUEIROZ; CURY, 2000) define uma abordagem que também projeta a construção de um supervisor para cada especificação  $i \in I$  envolvida no sistema, assim como o CMC. A diferença é que cada supervisor é obtido a partir de uma versão parcial da planta, e, logo, exerce a função de um *supervisor local*. A Figura 3 apresenta a arquitetura do CML (QUEIROZ; CURY, 2000).

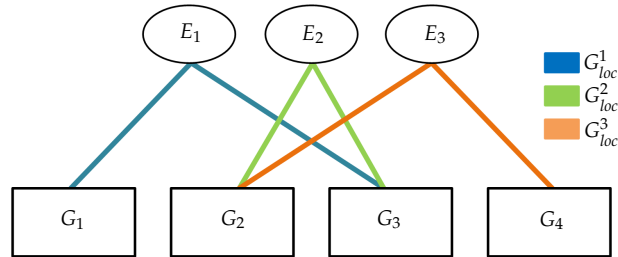


**Figura 3:** Arquitetura de controle CML.

Neste modelo, as plantas locais são denotadas por  $G_{loc}^i$  e são construídas por composições específicas de modelos de subsistemas. Quando um evento  $\sigma \in \Sigma$  é observado localmente, o respectivo supervisor  $S_{loc}^i$  mapeia um subconjunto de eventos  $S_{loc}^i(s) \subseteq \Sigma$  a serem habilitados na planta local. Eventos possíveis em  $G_{loc}^i$  que não fazem parte de  $S_{loc}^i(s) \subseteq \Sigma$  são inibidos pela ação de controle.

A escolha por quais plantas devem ser compostas localmente é regida pela intersecção entre os alfabetos dos modelos de cada subsistema e o de uma especificação em particular, ou seja, para cada especificação haverá um supervisor local composto

pela especificação correspondente e as plantas às quais se relaciona. A Figura 4 exemplifica a composição de plantas locais para um caso específico envolvendo os seguintes parâmetros:  $m = 3$  e  $n = 4$ , onde  $m$  denota o número de especificações e  $n$  denota o número de modelos de plantas assíncronas.



**Figura 4:** Formação de supervisores locais no CML.

Na Figura 4, a conexão entre as especificações e as plantas, representadas pelas linhas coloridas, denotam o compartilhamento de eventos entre os modelos envolvidos. As plantas locais  $G_{loc}^1$ ,  $G_{loc}^2$  e  $G_{loc}^3$ , construídas respectivamente para as especificações  $E_1$ ,  $E_2$  e  $E_3$ , são compostas apenas pelas plantas que possuem eventos em comum com a especificação correspondente. Formalmente, esta ideia pode ser apresentada como:

Por definição (NOURELFATH; NIEL, 2004): *Para um SED cuja planta é modelada por autômatos assíncronos  $G^j$ ,  $j \in J = 1, \dots, n$ , definidos em  $\Sigma^{G^j}$ , seja  $E^i$ ,  $i \in I = 1, \dots, m$  um conjunto de especificações modeladas em  $\Sigma^{E^i}$ . Para cada  $E^i$ , defina uma planta local  $G_{loc}^i$ , tal que:*

$$i \in j$$

$$G_{loc}^i = \parallel_{j \in J_i} G^j$$

$$J_i = \{j \in J \mid \Sigma^{G^j} \cap \Sigma^{E^i} \neq \emptyset\}$$

O CML define, então, o uso de  $G_{loc}^i$  na síntese de supervisores locais, e o *Problema de Controle Supervisório Modular Local (PCS-ML)* reintroduz PCS incorporando essa ideia: *Dada uma planta local  $G_{loc}^i$  para uma especificação  $E^i \subseteq \Sigma^{E^i} \subseteq \Sigma^*$ , definindo um comportamento desejado  $K_{loc}^i = E^i \cap L^w(G_{loc}^i)$ , encontre um supervisor local não-bloqueante  $S_{loc}^i$ , tal que  $L^w(S_{loc}^i/G_{loc}^i) \subseteq K_{loc}^i$ .*

Dois aspectos relevantes surgem no contexto PCS-ML. O primeiro é a necessidade de se conhecer as implicações do uso de uma planta local (parcial) para obter um

supervisor  $S_{loc}^i$ , em substituição de uma planta monolítica. O segundo aspecto remete à relação global de equivalência entre o conjunto de soluções locais para o PCS-ML e uma solução monolítica para o PCS.

O conjunto de supervisores locais gerados por meio da abordagem CML pode resolver o PCS original. Porém, eventualmente é possível haver divergências entre eles devido ao fato de que cada supervisor reconhece apenas uma parte da planta. Por este motivo, é necessário que o modelo do supervisor seja checado em relação à propriedade de *conflito* (QUEIROZ; CURY, 2000), para se certificar de que cada ação de controle é validada por todos os supervisores locais. Isso é absorvido pela intersecção na Figura 3. Se qualquer supervisor desabilita uma ação, então ela é desabilitada por todos. Tal checagem é uma operação automática implementada pela maioria das ferramentas de modelagem de SEDs. Caso não haja conflito, a ação dos supervisores modulares é equivalente à do supervisor monolítico, muito embora o procedimento de síntese seja substancialmente menos complexo, conforme avaliado a seguir.

#### 2.4.2.1 COMPLEXIDADE DA SÍNTESE MODULAR LOCAL

Seja um SED modelado por  $p$  subsistemas com no máximo  $m$  estados cada. Então,  $m^p$  é o número máximo de estados para o autômato que modela a planta. Ainda, seja esse sistema restrito por  $e$  especificações modeladas com no máximo  $n$  estados. Como cada especificação restringe uma parte  $k$  dos  $p$  subsistemas, então o cálculo de um supervisor modular local tem complexidade na ordem de

$$O[(n.m^k)^2].$$

Comparativamente, no CMC a complexidade é na ordem de

$$O[(n.m^p)^2]$$

e, no caso monolítico, na ordem de

$$O[(n^e.m^p)^2].$$

Claramente, a síntese no CML tende a ser computacionalmente mais vantajosa do que no CMC e, conseqüentemente, mais do que no procedimento monolítico. De fato, o fator de complexidade  $p$  é diretamente dependente do número de subsistemas envolvidos no problema, enquanto que o fator  $k$  varia em razão do número de

subsistemas afetados pela especificação.

A partir da TCS, todo sistema que possui uma dinâmica a eventos discretos pode ser modelado por estruturas computacionalmente processáveis (finitas). Dessa maneira, é possível sintetizar um controlador baseado na TCS para o processo avícola, que neste contexto é considerado um SED.

Inicialmente, é necessário definir uma estrutura física real para galpões de frangos de corte, e, partindo desta arquitetura, fazer o levantamento de cada componente do controle de ambiência (sensores e atuadores) para dar início ao desenvolvimento dos modelos que descrevem o sistema. A seção seguinte apresenta possíveis arquiteturas e seus componentes para aviários, fornecendo assim um arranjo a ser adotado para a obtenção de um controlador.

## 2.5 PROCESSO AVÍCOLA

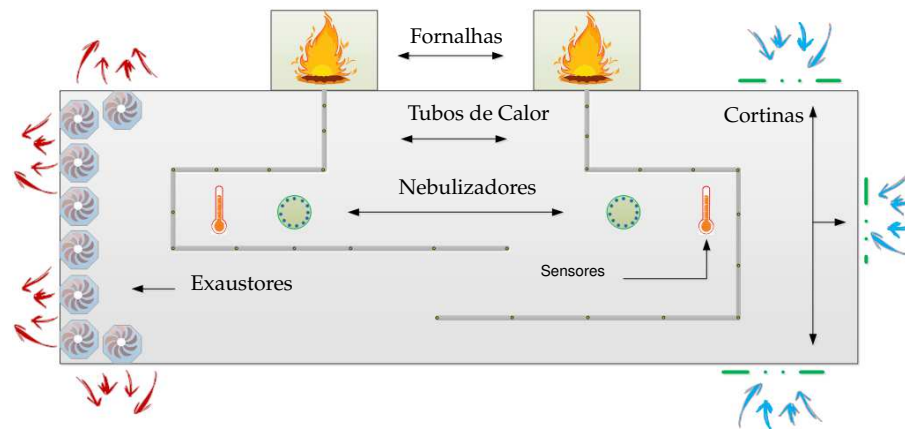
Nesta seção são discutidos os principais aspectos físicos dos processos avícolas. Duas arquiteturas diferentes sobre as quais um processo avícola pode ser conduzido são apresentadas, uma com fluxo de ar longitudinal e outra com fluxo de ar transversal, juntamente com suas características quanto à harmonização climática. Com base nestas estruturas e componentes, o processo avícola passa a ser modelado utilizando as abordagens da TCS.

### 2.5.1 ARQUITETURA COM FLUXO DE AR LONGITUDINAL

Segundo padrões e dados apresentados por órgãos como EMBRAPA (Empresa Brasileira de Pesquisa Agropecuária) e FAEP (Federação da Agricultura do Estado do Paraná), um aviário é uma construção retangular, cujas dimensões variam entre 100 e 200 metros de comprimento e entre 10 e 16 metros de largura, onde são criadas aves de diferentes espécies, em sua maioria, frangos de corte.

Um modelo tradicional de criação de frangos de corte é apresentado a seguir, onde os componentes internos do aviário são dispostos como na Figura 5. Este modelo faz parte da maioria dos padrões para construção de galpões definidos para criação de frangos de corte para diversas empresas brasileiras e é muito comum na região Sul do país.

O sistema convencional de controle de temperatura e umidade baseia-se nos



**Figura 5:** Arquitetura com fluxo de ar longitudinal

sinais de leitura de, no máximo, um par de sensores distribuídos ao longo do aviário, tomando decisões em conformidade com as necessidades climáticas. As decisões correspondem à ativação de atuadores físicos com o objetivo de alterar as condições climáticas internas de acordo com os parâmetros estabelecidos pelo avicultor.

Três atuadores desempenham um papel fundamental no equilíbrio térmico: *exaustores*, *nebulizadores* e *aquecedores*. Aquecedores são tubos que conduzem calor produzido externamente, geralmente por combustão em fornos, tornando o ambiente mais quente (BREWER et al., 1981; ZHU; LEE, 2005). Exaustores são equipamentos que removem ar de dentro para fora. Em casos mais extremos de calor, os nebulizadores vaporizam água no ambiente para fins de resfriamento.

Nesta arquitetura, os exaustores se localizam em uma das extremidades do aviário, enquanto as entradas de ar se localizam na extremidade oposta. Assim, a renovação de ar gera um fluxo linear que percorre toda sua extensão longitudinal, o que resulta em uma disfunção no equilíbrio climático pois, naturalmente, a ação de exaustão é mais intensa nas proximidades dos equipamentos do que na extremidade oposta.

Outro efeito imediato dessa arquitetura é que os exaustores atuam ao longo do comprimento do aviário. Assim, por exemplo, quando um exaustor posicionado em uma extremidade é acionado, é esperado que o fluxo de ar ocorra de maneira intensa em suas proximidades, desfavorecendo a exaustão da zona lateral oposta.

Para promover o aquecimento das aves, embora existam outras opções, o método mais utilizado nos dias frios é o das fornalhas à lenha e é o mais apropriado, pois oferece boas condições térmicas para os frangos, sendo uma alternativa mais barata para o produtor, uma vez que ele pode fornecer a madeira através de projetos

de reflorestamento (CATELAN, 2007; FUNCK; FONSECA, 2007). As fornalhas à lenha fornecem calor a tubos condutores, e estes se encarregam da distribuição de ar quente ao longo do aviário por meio de aberturas. Em dias de frio, ou mesmo nos primeiros dias de vida das aves, é necessário manter o fogo aceso para aquecer o ambiente.

Em alguns modelos são utilizadas campânulas à gás distribuídas ao longo do aviário, consideradas eficientes e que favorecem sistemas de acionamento automático. Porém, atualmente são pouco utilizadas devido ao consumo excessivo de gás, tornando-se inviável financeiramente à longo prazo. O sistema de aquecimento a gás é considerado mais custoso financeiramente do que os sistemas à lenha. Mesmo assim continua sendo usado em alguns casos, por não ser necessário o reabastecimento manual frequente, principalmente durante a noite (CATELAN, 2007).

Outro exemplo de sistema de aquecimento são tambores de irradiação a lenha, utilizados geralmente para suplementar o aquecimento proporcionado pelas campânulas de gás. Trata-se de um aquecedor por radiação, que emite calor no ambiente ao redor de onde estiver localizado.

Sob o efeito de dispositivos de resfriamento ou aquecimento, a renovação do ar dentro deste aviário gera um fluxo linear, o que normalmente provoca disfunções no equilíbrio climático porque a ação de exaustão é naturalmente mais intensa nas proximidades do equipamento do que na extremidade oposta. Especialmente em regiões onde o tempo frio predomina, como é o caso do Sul do Brasil, torna-se um desafio alcançar o conforto térmico em condições tão heterogêneas (VIEIRA et al., 2011). Na prática, o avicultor percebe e reage a todo o controle do processo. Nesses dias, se por um lado é necessário manter o conforto térmico das aves (desligando a exaustão ao máximo e produzindo calor), por outro lado é fundamental eliminar concentrações de amônia<sup>1</sup> (ligando exaustão ao mínimo), a fim de garantir a qualidade do ar. Portanto, encontrar o equilíbrio exato de conforto é complexo.

Na prática, o avicultor percebe e reage a todo o controle do processo, que é ineficiente, desconfortável e sujeito a erros.

Outra exigência técnica torna o controle do processo ainda mais complexo: Se por um lado é necessário manter o conforto térmico (arrefecimento ou aquecimento, conforme necessário), por outro lado é essencial eliminar as concentrações exageradas de amônia, um gás incolor gerado a partir da decomposição microbiana dos dejetos

---

<sup>1</sup>Amônia é um gás incolor gerado a partir da decomposição microbiana de dejetos, que causa significativas perdas econômicas na criação das aves (HARPER et al., 2010).

animais que provoca significativas perdas econômicas se não controlada (LIN et al., 2016; HARPER et al., 2010). A eliminação de amônia exige que a exaustão seja sempre ativa, o que contradiz uma eventual necessidade de aquecimento, por exemplo. Portanto, encontrar o equilíbrio de conforto certo é complexo, em geral.

Alternativas para abordar tais problemas são sinalizadas na literatura. Por exemplo, outros modelos de arquiteturas para aviários (XIN et al., 2013; BUSTAMANTE et al., 1994) foram considerados considerando readaptações na estrutura, como promover a entrada e saída de ar nas laterais, ou seja, com fluxo transversal e não mais longitudinal. Além disso, outras alternativas para coleta de dados foram aplicadas, como por exemplo redes de sensores, a fim de coletar amostras mais confiáveis e abrangentes de condições climáticas (SALLABI et al., 2011; GAO et al., 2016; SOLDATOS et al., 2005). Essas novas perspectivas sugerem que o *layout* de aviários pode ser explorado para melhorar o controle automático.

## 2.6 ARQUITETURA COM FLUXO DE AR TRANSVERSAL

Existem variações entre os modelos adotados para galpões de aves de corte em relação à direção do fluxo de ar, aos tipos de equipamentos para arrefecimento, aquecimento e iluminação e também os tipos de controladores utilizados.

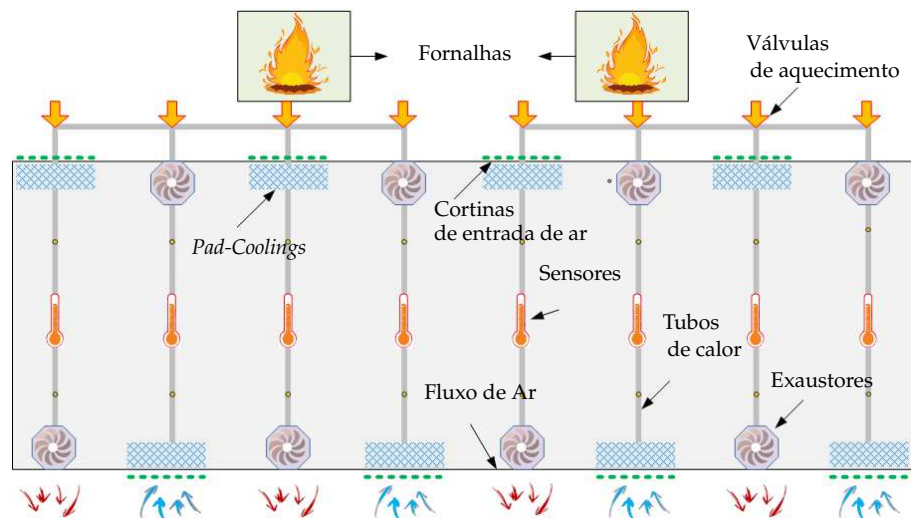
Uma alternativa é apontada por tecnologias emergentes para aviários (BICHARA, 2011), que considera promover a entrada e a saída lateral de ar e a coleta das condições climáticas por meio de redes de sensores (LOPES, 2015) (VIEIRA, 2009). No entanto, nesse caso, a estrutura de controle requer escalabilidade para absorver esses novos sinais e novos requisitos.

Com base nas desvantagens observadas no formato aviário com fluxo de ar longitudinal apresentado, este trabalho baseia-se em uma arquitetura de ventilação transversal.

Neste modelo, os nebulizadores são substituídos pelo componente *pad-cooling*, uma alternativa de refrigeração que consiste em uma placa com superfície evaporativa que tem se tornado comum entre as empresas de produção avícola. O *pad-cooling* é responsável por reduzir a temperatura do ambiente por meio da umidificação por evaporação de água. Os painéis evaporativos utilizados nesse processo são geralmente de material especial de celulose, mantidos constantemente umedecidos, através do qual o ar passa e resfria-se antes de entrar no aviário, diferenciando-se assim, dos

nebulizadores convencionais, que "borrifam" água em suas proximidades. É recomendado que não seja utilizada nebulização interna, como a apresentada na arquitetura tradicional, em um galpão que já possui um sistema de *pad-coolings*, pois ambos em conjunto causariam aumento significativo e indesejado da umidade relativa do ar.

O modelo de aviário com ventilação transversal, ou em *zig-zag*, é ilustrado na Figura 6. Esta arquitetura é uma das variações possíveis para galpões de frangos de corte (KREVIINGHAUS et al., 1996; BUSTAMANTE et al., 1994; BLANES-VIDAL et al., 2008, 2007).



**Figura 6:** Arquitetura com fluxo de ar transversal.

Observa-se que, essencialmente, os mesmos equipamentos utilizados pelo modelo anterior são reorganizados internamente, com a única exceção do nebulizador, que foi substituído pelo *pad-cooling*, considerando uma relação lateral entre entradas de ar, exaustores, nebulizadores, rede de sensores e tubos de aquecimento.

Além disso, é notável que o processo se torna equivalentemente segmentado de acordo com o fluxo de ar transversal. Assume-se que cada setor recebe um conjunto completo de dispositivos, incluindo: uma entrada de ar, um exaustor, um *pad-cooling*, um ponto de aquisição de dados (sensores) e um aquecedor local, que abre uma válvula através da qual o ar quente entra no setor pelo condutor de calor correspondente.

Esta arquitetura foi adotada como um modelo base para a obtenção do controlador para processos avícolas por possuir características que permitem prover maior número de sensores, por possibilitar que as ações de controle sejam feitas em cada setor isoladamente, entre outros benefícios.

O objetivo é implementar uma estrutura de controle para coordenar todos esses



equipamentos em paralelo, o que será abordado na próxima seção. Primeiramente, o modelo de arquitetura com fluxo de ar transversal é adotado e, sobre ele, são realizadas as etapas de modelagem, síntese e implementação do controlador.

### 3 MÉTODO

Este capítulo aborda os métodos aplicados no processo de desenvolvimento do controlador para processos avícolas proposto. São apresentados os modelos da planta e especificações para cada subsistema do processo avícola.

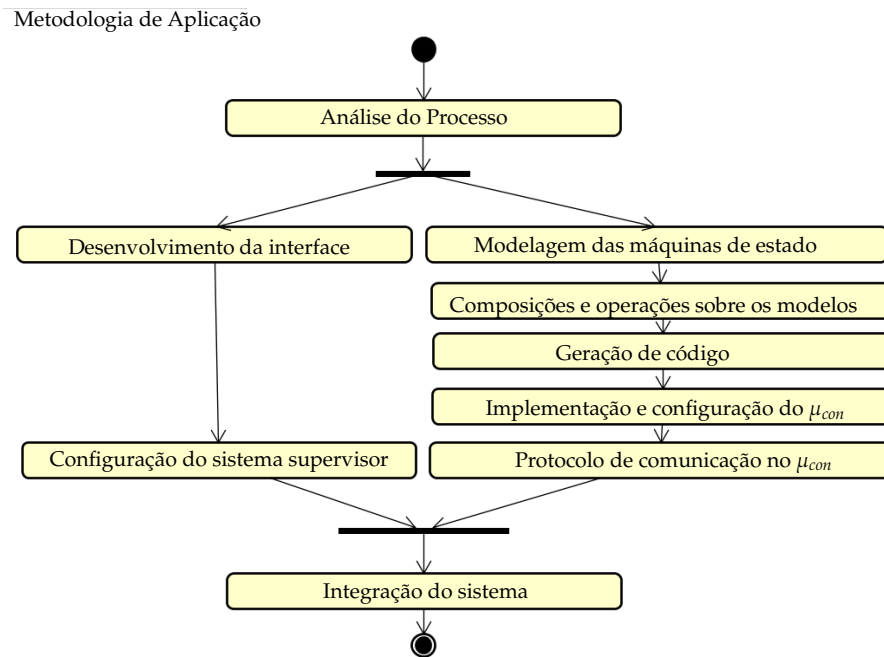
#### 3.1 MÉTODO DE MODELAGEM, CONTROLE E SUPERVISÃO

Esta seção descreve como cada componente de um processo avícola pode ser modelado dentro da arquitetura de aviário adotada.

A ideia básica é coordenar apropriadamente os atuadores. Os atuadores são os dispositivos que conectam espaços físicos e virtuais. Os sinais são coletados por sensores no ambiente e fornecidos à estrutura de controle, que então toma decisões e os entrega, também sob a forma de sinais, aos atuadores, que são então ativados, realizando ações físicas de controle, alterando a condição climática do ambiente. Assim, para que essas ações de controle sejam possíveis, tem-se basicamente cinco tarefas a serem realizadas:

- (i) Obter um modelo para cada componente do sistema (*modelo de planta*), incluindo sensores, exaustores, *pad-coolings*, cortinas de ar e aquecedores;
- (ii) As regras do projeto devem ser cumpridas sob controle (*modelos de especificação*);
- (iii) Combinar plantas e modelos de especificação com o objetivo de sintetizar um modelo para o controlador (*síntese de controle*);
- (iv) Traduzir o modelo de controlador para código implementável (*implementação do controlador*);
- (v) Implementar um sistema de supervisão para o processo a ser monitorado sob controle (*monitoramento do processo*).

Considerando que a aplicação possui múltiplos passos desde a definição e modelagem de componentes, até a implementação de hardware e monitoramento de processos, é proposta na sequência uma metodologia que orienta essas etapas. As principais tarefas a serem seguidas são organizadas e expressadas por um diagrama, apresentado na Figura 7. As próximas seções abordam individualmente este conjunto completo de tarefas.



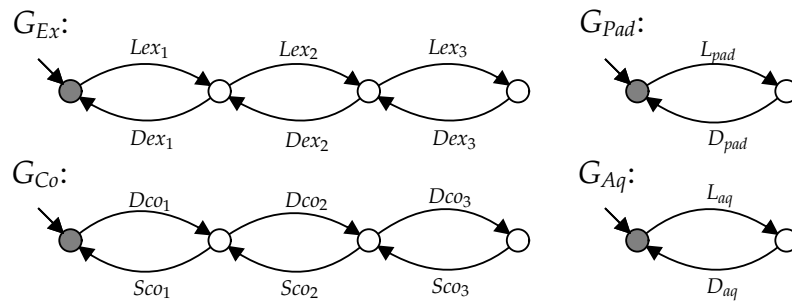
**Figura 7:** Método que conduz a uma ferramenta de monitoramento para um processo controlado de avicultura

## 3.2 MODELAGEM DA PLANTA DO SISTEMA POR AUTÔMATOS FINITOS

Por simplicidade e objetividade, os modelos a serem apresentados consideram uma única faixa transversal de ventilação. Naturalmente, tais modelos podem ser replicados para qualquer quantidade de faixas sem perda de generalidade.

### 3.2.1 MODELAGEM DE ATUADORES

A Figura 8 apresenta os modelos para cada dispositivo atuador que compõe o sistema, os quais são explanados em seguida.



**Figura 8:** Modelos de atuadores

### 3.2.1.1 MODELO DO EXAUSTOR

O autômato  $G_{Ex}$  na Figura 8 é um modelo genérico para um exaustor de três níveis de potência. São considerados três níveis de operação em vez de um pois, na prática, isso pode ser eficiente em termos energéticos, além de evitar o equipamento de ser ligado e desligado freqüentemente. Por outro lado, pode-se sugerir mais de 3 fases de operação (MIRZAEI-GHALEH et al., 2015). Nesse caso, para uma ação de controle mais fragmentada, uma abordagem de *controle contínuo* (OGATA, 2003) poderia ser mais efetiva. Para o propósito deste trabalho, argumenta-se que a discretização do comportamento de um exaustor em três fases é suficiente para o controle. É possível afirmar pois, ao monitorar aviários reais em plenas condições de funcionamento, nota-se que as condições climáticas internas não mudam tão rapidamente a ponto de não serem seguidas efetivamente pelo controlador.

Outros argumentos a favor do controle de três fases incluem: evitar mudanças súbitas na circulação de ar; facilidade de implementação de políticas de ativação mínima e máxima; etc. Observe que a ventilação mínima é essencial em dias frios, para a eliminação de amônia. Por outro lado, ventilação extrema é importante para retardar a ativação de nebulizadores, que reduzem a temperatura bruscamente, mas interferem diretamente no teor de umidade, o que pode ser prejudicial para as aves.

Assim, neste trabalho, assume-se que um exaustor opera em até três níveis de ventilação, sendo eles: 1: *mínimo*; 2: *moderado*; e 3: *máximo*, como mostra o autômato  $G_{Ex}$  na Figura 8. Os eventos  $Lex_i$  e  $Dex_i$  denotam, respectivamente, transições de *início* e *fim* de/para as fases correspondentes  $i = 1, 2, 3$ . O comportamento modelado em  $G_{Ex}$  é de comutação sequencial, exatamente como se espera do equipamento real, na prática.

### 3.2.1.2 MODELO DO PAD-COOLING

Na prática, um *pad-cooling* é ativado somente em casos extremos, quando o sistema de exaustão alcança seu máximo poder de atuação e ainda assim a temperatura não diminui até o parâmetro desejado. Então, para o modelo do comportamento de um *pad-cooling*, apenas dois estados são considerados: *ativo* e *inativo*, como mostra o autômato  $G_{Pad}$  na Figura 8. Em  $G_{Pad}$ , o único estado de parada é o estado inicial e os eventos do modelo possuem a seguinte notação:

Eventos  $L_{pad}$  e  $D_{pad}$  denotam, respectivamente, *início* e *fim* de operação. O comportamento modelado em  $G_{Pad}$  segue uma política *on/off*, exatamente como se espera do equipamento real.

### 3.2.1.3 MODELO DA CORTINA

A cortina é o componente que regula a entrada de ar externo no aviário. Cada cortina localiza-se no lado oposto ao exaustor do mesmo setor. Quando é sintonizada com o exaustor correspondente, a cortina desempenha um papel essencial para a renovação do ar dentro do aviário, permitindo que o ar puro entre enquanto o exaustor expulsa ar rico em amônia, promovendo a purificação do ambiente.

$G_{Co}$ , na Figura 8, apresenta um modelo para uma cortina. O autômato  $G_{Co}$  segue a mesma ideia do modelo  $G_{ex}$ . Ele foi planejado para ser controlado de acordo com  $G_{ex}$ , isto é, quando o exaustor é acionado, a cortina também é acionada seguindo a política de níveis de abertura (mínimo, intermediário e máximo). Nesse sentido, o modelo  $G_{Co}$  também é baseado em níveis, de modo que os eventos  $Lco_i$  e  $Dco_i$  respectivamente denotam transições *início* e *fim* de/para o nível de abertura correspondente  $i = 1, 2, 3$  da cortina. Para fins práticos, assume-se que, no estado inicial, a cortina está totalmente fechada, de modo que ela se abre progressivamente com os eventos  $Lco_i$  e fecha com os eventos  $Dco_i$ .

### 3.2.1.4 MODELO DO AQUECEDOR

Cada setor de ventilação contém uma válvula de fluxo de ar quente que pode ser aberta ou fechada. Este equipamento é integrado a uma tubulação geral que conduz o ar quente ao interior do aviário. A distribuição é por setor e é ativada pelo disparo de eventos provocados por sinais de sensores. Quando uma válvula é aberta, permite

a passagem de ar, que aquece o respectivo setor, principalmente.

$G_{Aq}$  na Figura 8 mostra um autômato de dois estados que modela o comportamento de um aquecedor. Em  $G_{Aq}$ , os eventos  $L_{aq}$  e  $D_{aq}$ , denotam, respectivamente, *início* and *fim* de operação. O comportamento modelado em  $G_{Aq}$  segue uma política *on/off*, exatamente como se espera do equipamento real.

### 3.2.2 MODELAGEM DOS SENSORES

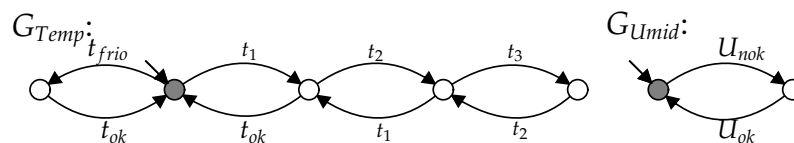
A partir da composição

$$G_{Ex} || G_{Pad} || G_{Co} || G_{Aq}$$

obtém-se a ação integrada de componentes em um processo avícola. Isso corresponde a um modelo que inclui todas as possíveis sequências de operações. Obviamente, tal modelo em malha aberta tem de ser controlado posteriormente.

Para que esse modelo seja controlado, depende dos eventos responsáveis pelo disparo das ações respectivas de cada dispositivo no sistema. Estes eventos são sinais periodicamente coletados por sensores no início do processo, que indicam níveis de temperatura e umidade. Eles fornecerão efetivamente os dados sobre os quais o sistema de controle será baseado para tomada de decisão.

Neste trabalho, são considerados dois sensores em cada setor: temperatura e umidade. Seus modelos estão presentes na Figura 9.



**Figura 9:** Sensor models

Os modelos  $G_{Temp}$  e  $G_{Umid}$  serão discutidos a seguir.

#### 3.2.2.1 MODELO DO SENSOR DE TEMPERATURA

$G_{Temp}$  na Figura 9 apresenta o modelo do sensor de temperatura. No contexto deste trabalho, este dispositivo está restrito à leitura ou detecção de quatro faixas, ou níveis, de temperatura possíveis: 1 nível para frio e 3 níveis para temperaturas quentes. Estes intervalos de temperatura são parâmetros que dependem da linhagem

dos frangos e das especificações para cada tipo de animal e de galpão. Por este motivo, são tratados em termos genéricos e, na prática, serão configurados de acordo com o caso real.

Poderiam ser considerados mais ou menos níveis, sem perda de generalidade. Esta escolha baseia-se na prescrição prática do processo, que recomenda que a temperatura seja mantida entre um determinado intervalo de valores inteiros, normalmente entre 25 e 35 graus *Celsius*, variando de acordo com o ciclo de vida do frango e de acordo com uma linhagem (CASSUCE et al., 2013, 2001).

Assim, neste modelo, foi considerado que o nível frio é atingido quando a temperatura coletada está abaixo do mínimo recomendado, o que é comum nos primeiros dias de vida das aves ou em dias de inverno. Esta situação é modelada usando o evento  $t_{frio}$  do estado inicial. Os sistemas permanecem lá até que as medidas de aquecimento sejam tomadas e pode retornar ao intervalo de temperatura aceitável (evento  $t_{ok}$ ), onde as aves estão em um estado de conforto térmico.

Outra situação é quando o sensor detecta um valor acima do parâmetro alvo. Neste caso, o evento  $t_1$  força uma transição para um estado no qual a temperatura pode ser restaurada para o intervalo de conforto (evento  $t_{ok}$ ), ou pode continuar a subir (eventos  $t_2$  e  $t_3$ ). A transição entre estados, em ambas as direções, ocorre quando a temperatura atinge os limites de intervalo definidos para cada evento  $t_{ok}$ ,  $t_1$ ,  $t_2$  e  $t_3$ .

Observe que, após um evento  $t_3$ , não há mais recursos exaustivos disponíveis para o resfriamento. Neste caso, considerando que a temperatura continua a aumentar, o nebulizador tem de ser ativado para maior refrigeração do local.

### 3.2.2.2 MODELO DO SENSOR DE UMIDADE

Por ser um ambiente fechado, controlado e sujeito a ventilação contínua, o excesso de umidade em frangos de corte, em geral, não conta com uma ação preventiva. A baixa umidade, por outro lado, é geralmente associada ao calor, alterando o comportamento dos frangos de corte (LIN et al., 2005).

Nesse sentido, para a modelagem do sensor de umidade, foi considerado que é suficiente observar no mínimo dois eventos, sendo um evento que se refere a um índice de umidade aceitável, e outro evento que aponta para um valor de umidade baixo, que sob controle permite ativar o *pad-cooling*, e garantir que este dispositivo não ocasione um aumento exagerado de umidade relativa, já que é utilizado essencialmente para

diminuir a temperatura.

$G_{Umid}$  na Figura 9 mostra um autômato de 2 estados que modela o comportamento de um sensor de umidade. Em  $G_{Umid}$ , os eventos  $U_{nok}$  e  $U_{ok}$  indicam, respectivamente, que a umidade está abaixo e dentro do parâmetro aceitável.

### 3.2.3 COMPOSIÇÃO DA PLANTA

Para a modelagem dos componentes (subsistemas) que integram o processo avícola, a controlabilidade dos eventos foi assumida tal que  $\Sigma_c = \{Lex_1, Lex_2, Lex_3, Dex_1, Dex_2, Dex_3, L_{pad}, D_{pad}, Lco_1, Lco_2, Lco_3, Dco_1, Dco_2, Dco_3, L_{aq}, D_{aq}\}$  enquanto  $\Sigma_u = \{t_{ok}, t_1, t_2, t_3, t_{frio}, U_{ok}, U_{nok}\}$ .

Todos os eventos considerados na modelagem do sistema são observáveis, portanto os efeitos de possíveis eventos não observáveis para esta aplicação não são abordados.

Após a construção dos modelos individuais, eles são compostos para formar a planta  $G$  do sistema, tal que

$$G = G_{Ex} || G_{Pad} || G_{Co} || G_{Aq} || G_{Temp} || G_{Umid}.$$

$G$  corresponde a um autômato com 640 estados e 4864 transições e modela o comportamento do sistema em malha aberta. Para fechar a malha, especificações de controle são construídas para o sistema, da maneira mostrada a seguir.

### 3.3 MODELAGEM DAS ESPECIFICAÇÕES DO SISTEMA POR AUTÔMATOS FINITOS

Na TCS, especificações de controle <sup>1</sup> podem também ser modeladas através de autômatos e combinadas ao modelo de sistema por composição síncrona. Isto conduz a uma versão restrita do modelo de sistema, em que sequências indesejáveis são evitadas, isto é, um modelo de especificação impõe uma ação proibitiva em  $G$ .

No entanto, projetar uma especificação é uma tarefa criativa que não pode ser totalmente automatizada. Em contrapartida, as especificações podem envolver uma grande e intrincada combinação de estados e eventos, de modo que, em geral, a tarefa

---

<sup>1</sup>Consideramos que os estados de um modelo de especificação são todos marcados, de modo que a marcação final é responsável pelo modelo de planta.



é restrita a especialistas em lógicas de controle. No sentido de facilitar a concepção de especificações, apresenta-se um raciocínio sistemático que pode ajudar a modelar uma especificação.

### 3.3.1 DESIGN SISTEMÁTICO

Seja  $G$  um modelo de planta e  $E$  um modelo que pretenda restringir  $G$  a determinado comportamento especificado. Resta saber como  $E$  pode ser construído. Embora esta seja uma tarefa não sistemática, as etapas gerais que levam a  $E$  podem ser resumidas da seguinte maneira:

- (i) Descrever a versão textual da especificação;
- (ii) Interpretar a descrição da especificação para projetar como ela poderia ser construída;
- (iii) Construir um autômato que expressa rigorosamente todos e apenas os requisitos definidos na versão textual.

Observe que essas etapas são puramente tarefas de engenharia. Foi descrito na literatura que, em alguns casos, tais tarefas podem tornar-se complexas, particularmente quando envolvem a dependência de dados, que pode então ser tratada por extensões de modelagem específicas (CURY et al., 2015; TEIXEIRA et al., 2014).

Neste trabalho, assume-se que as etapas (i) e (ii) estão bem definidas a partir da percepção do engenheiro e, em seguida, é fornecido um guia para auxiliar na tarefa (iii), ou seja, na construção de uma própria especificação.

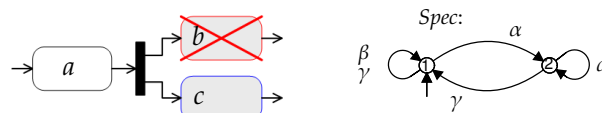
Seja  $G$  o modelo da planta e  $\mathcal{E} = \{\mathcal{E}^i\}$ ,  $i \in I = \{1, \dots, n\}$ , um conjunto de descrições de especificações para  $G$ , que são modeladas por  $E = \prod_{i \in I} E^i$ ,  $i \in I$ . A construção de cada  $E^i$  pode ser descrita aproximadamente como segue:

- (a) identifique a ação  $\sigma$  a ser proibida na planta. Em termos de modelagem, isso significa identificar o evento  $\sigma \in \Sigma^G$  a ser desabilitado. Por uma questão de simplicidade e modularidade, é conveniente desativar um único evento para cada modelo de especificação;
- (b) se uma determinada ação  $\sigma$  deve ser evitada no sistema, ela provavelmente tem uma (ou múltiplas) ações antecedente  $\sigma'$ . Portanto, também pode ser identificado

de forma equivalente o conjunto de eventos  $\Sigma^p \subseteq \Sigma^G$  tal que cada  $\sigma' \in \Sigma^p$  precede imediatamente  $\sigma$ ;

- (c) construa um autômato  $E^i$ , correspondente aos requisitos em  $\mathcal{E}^i$ , tal que, depois de  $\sigma'$ , desabilita  $\sigma$ ;
- (d) habilite  $\sigma'$  em todos os outros estados  $q \in Q_{E^i}$ .

A Figura 10 mostra um exemplo de modelagem de especificação. Considere uma planta composta por uma determinada tarefa  $a$  (modelada por um evento  $\alpha$ ), seguida por duas tarefas  $b$  (modeladas por um evento  $\beta$ ) e  $c$  (modeladas por um evento  $\gamma$ ). Suponha que o objetivo de controle é evitar que  $b$  ocorra após a ação  $a$ . Para expressar essa restrição, projeta-se um modelo  $Spec$  que habilita  $\beta$  no estado inicial e o proíbe após um evento  $\alpha$ , como mostrado na Figura 10.



**Figura 10:** Exemplo de modelagem de especificação

Observe que  $\gamma$  e  $\alpha$  são habilitados em ambos os estados 1 e 2. Como  $\alpha$  e  $\gamma$  não são alvos para restrição, eles devem ser ativados em todos os estados. Talvez eles não fossem habilitados pelo modelo de sistema de qualquer maneira, mas isso não é uma preocupação ao projetar a especificação. Por outro lado, o evento  $\beta$  é alvo de restrição. Portanto, para ser proibido após  $\alpha$ , ele é incluído no auto-laço do estado inicial. Como  $\beta \in \Sigma^{Spec}$  e  $1 \xrightarrow{\alpha} 2 \not\rightarrow \beta$ , então a especificação foi projetada corretamente.

Algumas observações são importantes neste ponto. Em primeiro lugar, ao projetar uma especificação, a controlabilidade dos eventos não é levada em conta, pois esta é responsável pelo próprio algoritmo de síntese. Além disso, vale a pena lembrar que as cadeias pertencentes ao comportamento do sistema não necessariamente devem ser rastreadas pelo autômato de especificação a partir do estado inicial. Na verdade, os traços possivelmente reconhecidos pelo sistema já são considerados elegíveis no modelo do sistema.

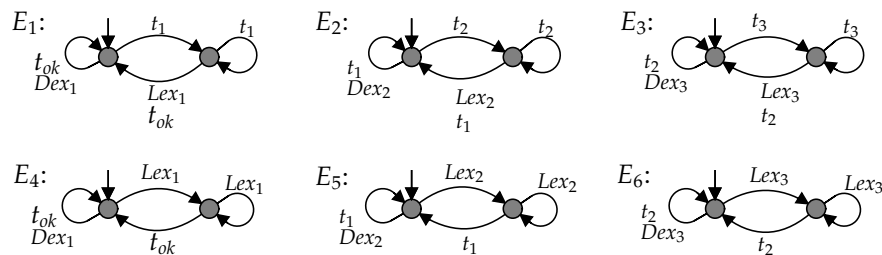
Estruturas similares a  $Spec$  são usadas para evitar os problemas mais gerais de coordenação em plantas de processo avícola.

### 3.3.1.1 COORDENAÇÃO DO EXAUSTOR

Para coordenar um equipamento exaustor é necessário, basicamente, controlar as ações de ativação e desativação, que estão relacionadas às seguintes descrições de controle:

- $E_1$  - Proíba ligar o primeiro nível do exaustor ( $Lex_1$ ) enquanto a temperatura não atingir o nível  $t_1$ ;
- $E_2$  - Proíba ligar o segundo nível do exaustor ( $Lex_2$ ) enquanto a temperatura não atingir o nível  $t_2$ ;
- $E_3$  - Proíba ligar o terceiro nível do exaustor ( $Lex_3$ ) enquanto a temperatura não atingir o nível  $t_3$ ;
- $E_4$  - Proíba o desligamento do exaustor enquanto a temperatura não atingir o nível  $t_{ok}$ ;
- $E_5$  - Proíba o desligamento do segundo nível do exaustor enquanto a temperatura não atingir o nível  $t_1$ ;
- $E_6$  - Proíba o desligamento do terceiro nível do exaustor enquanto a temperatura não atingir o nível  $t_2$ ;

A Figura 11 mostra os modelos de  $E_i$ ,  $i = 1, \dots, 6$ .



**Figura 11:** Modelos de especificação para a operação do exaustor

As três primeiras especificações,  $E_1$ ,  $E_2$  e  $E_3$ , destinam-se a controlar a progressão do exaustor, enquanto  $E_4$ ,  $E_5$  e  $E_6$ , controlam a operação inversa. Por exemplo, o autômato  $E_1$  desativa o evento  $Sex_1$  no estado inicial e somente o habilita após a ocorrência de um evento  $t_1$ . Em outras palavras, o exaustor só pode ser ativado quando a temperatura sobe para o primeiro nível. Uma ideia equivalente aplica-se aos modelos  $E_2$  e  $E_3$  para controlar os outros níveis de exaustão.

Por outro lado, o modelo  $E_4$  evita que um exaustor seja desligado enquanto a temperatura é maior do que a configuração. Para implementar essa ação,  $E_4$  proíbe o evento  $Dex_1$  no estado inicial e somente a habilita após um evento  $t_{ok}$ , o que garante que a temperatura medida é aceitável. Uma ideia equivalente aplica-se aos modelos  $E_5$  e  $E_6$ .

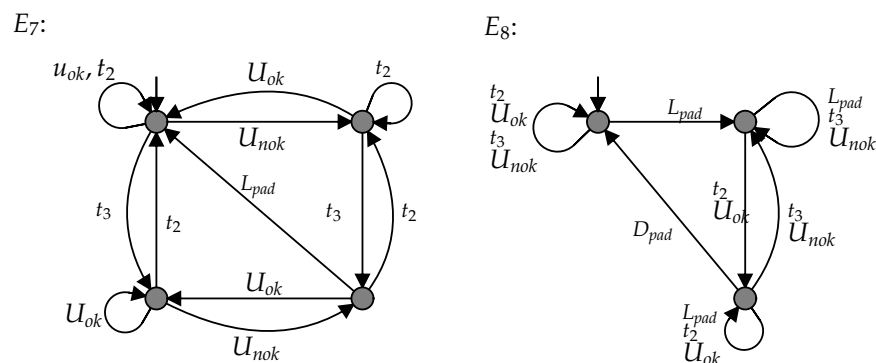
### 3.3.1.2 COORDENAÇÃO DO NEBULIZADOR

A coordenação do equipamento *pad-cooling* também segue basicamente as ações ativar e desativar, porém, um pouco mais complexo do que os modelos anteriores. De fato, um *pad-cooling* não é simplesmente controlado de acordo com um parâmetro recolhido por um sensor. Em vez disso, depende de uma combinação de fatores e regras que devem ser seguidas para não comprometer a qualidade do clima com excesso de umidade.

São propostas duas regras básicas para o funcionamento do *pad-cooling*:

- $E_7$  - Proíba o acionamento do *pad-cooling* enquanto a temperatura não atingir o nível  $t_3$  e a umidade estiver baixa ( $u_{nok}$ ), em qualquer ordem de ocorrência;
- $E_8$  - Proíba o desligamento do *pad-cooling* enquanto a umidade ainda estiver baixa e enquanto a temperatura ainda estiver muito alta. Permite apenas após ocorrer  $t_2$  e  $u_{ok}$ , sendo que as duas condições são necessárias.

A Figura 12 mostra os modelos para  $E_7$  e  $E_8$ .



**Figura 12:** Modelos de especificação para o funcionamento do *pad-cooling*

O autômato  $E_7$  modela uma lógica que proíbe  $L_{pad}$  enquanto dois eventos não ocorrem:  $u_{nok}$  e  $t_3$ . Isto significa que o *pad-cooling* não será ativado até que sejam

satisfeitas duas condições necessárias: A umidade é menor que o valor aceitável (em outras palavras, a umidade não está muito alta) e a temperatura está no nível extremo. Desta forma, a ativação do *pad-cooling* só ocorre em casos extremos de calor, exatamente como se espera que se comporte na prática.

O autômato  $E_8$  modela as condições para a desativação do *pad-cooling*. Ele proíbe o evento  $D_{pad}$  no estado inicial, e o habilita após qualquer uma das duas condições não for mais satisfeita. Ou seja, o *pad-cooling* é desligado imediatamente após a ocorrência de  $u_{ok}$  ou  $t_2$ , em qualquer ordem.

### 3.3.1.3 COORDENAÇÃO DA CORTINA

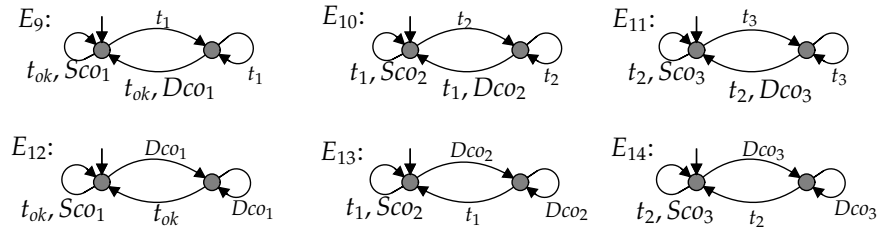
Coordenar uma cortina de aviário refere-se a controlar adequadamente a entrada de ar externo no ambiente. Tem-se, a partir da descrição do processo na seção 2.5, que as entradas de ar estão localizadas opostas ao exaustor do mesmo setor e que ambos os equipamentos trabalham em sincronia, ou seja, o ar a ser esgotado do aviário deve ser corresponde à entrada de ar, passando pelas aves para arrefecimento. Portanto, controlar uma cortina depende do controle do exaustor ou, de forma equivalente, do sensor que ativa o exaustor.

Em seguida, são apresentadas seis especificações relacionadas à coordenação da cortina. Referem-se basicamente a ações de abertura e fechamento, e sua integração ao comportamento do exaustor é promovida por um evento-chave: a leitura da temperatura.

- $E_9$  - Proíba a abertura do primeiro nível da cortina enquanto a temperatura não alcançar  $t_1$ .
- $E_{10}$  - Proíba a abertura do segundo nível da cortina enquanto a temperatura não atingir  $t_2$ .
- $E_{11}$  - Proíba a abertura do terceiro nível da cortina enquanto a temperatura não atingir  $t_3$ .
- $E_{12}$  - Proíba o fechamento total da cortina (ou primeiro nível) enquanto a temperatura não estiver em uma faixa aceitável  $t_{ok}$ .
- $E_{13}$  - Proíba o fechamento do segundo nível da cortina enquanto a temperatura não atingir  $t_1$ .

- $E_{14}$  - Proíba o fechamento do terceiro nível da cortina enquanto a temperatura não atingir  $t_2$ .

Um autômato para cada especificação é mostrado na Figura 13.



**Figura 13:** Modelos de especificação para o funcionamento da cortina

$E_9$ ,  $E_{10}$  e  $E_{11}$  na Figura 13 controla a abertura da cortina, enquanto  $E_{12}$ ,  $E_{13}$  and  $E_{14}$  coordenam a operação de fechamento.

Os autômatos  $E_9$ ,  $E_{10}$  and  $E_{11}$  proíbem a abertura da cortina no estado inicial (eventos  $Sc_{oi}$ ,  $i = 1, 2, 3$ ), e são complementados pelos autômatos  $E_{12}$ ,  $E_{13}$  e  $E_{14}$  que restringem seu fechamento (eventos  $Fco_i$ ,  $i = 1, 2, 3$ ) enquanto a temperatura estiver fora da faixa esperada (eventos  $t_i$ ,  $i = ok, 1, 2$ ).

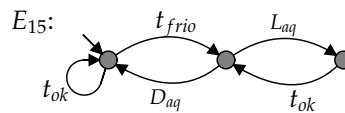
Essas especificações são semelhantes às apresentadas na Figura 11, para o controle do exaustor. De fato, ambos os equipamentos são coordenados de acordo com a temperatura coletada. Quando ela sobe ao primeiro nível, o exaustor é ativado para seu primeiro nível, e a cortina abre seu primeiro nível, assim, sucessivamente.

#### 3.3.1.4 AQUECEDOR

Controlar o equipamento responsável por aquecer o ambiente requer orquestrar as ações de ligar e desligar na válvula do aquecedor. Foi definida uma única especificação para descrever essa lei de controle:

- $E_{15}$  - Proíba a ativação do aquecedor enquanto a temperatura não estiver fria e, quando estiver, proíba o desligamento do aquecedor até que a temperatura esteja confortável novamente.

A Figura 14 propõe um modelo de especificação para a regra de operação do aquecedor.

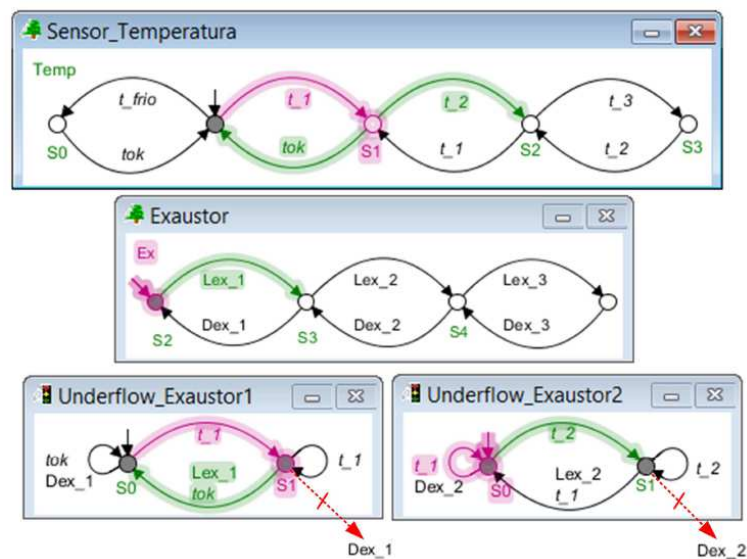


**Figura 14:** Modelo de especificação para acionamento e desligamento do aquecedor

O modelo  $E_{15}$  proíbe  $L_{aq}$  (ativação do aquecedor) no estado inicial, até a temperatura atingir um valor baixo, ou estiver fria (evento  $t_{frio}$ ) e proíbe o evento  $D_{aq}$  (desativação do aquecedor) caso contrário, até que a temperatura seja aceitável (evento  $t_{ok}$ ).

### 3.3.2 ESPECIFICAÇÃO GLOBAL

Para exemplificar de uma maneira geral o comportamento de uma especificação, pode-se observar a Figura 15, que mostra como uma ação proibitiva é interpretada graficamente pelo simulador do software Supremica, utilizando o exemplo real da aplicação apresentada.



**Figura 15:** Simulação de uma ação proibitiva no software Supremica por meio de especificações

Após a ocorrência de  $t_1$ , a especificação *Underflow\_Ex<sub>1</sub>* ( $E_1$ ) inibe a ocorrência de  $Dex_1$  até que ocorra  $Lex_1$  ou  $t_{ok}$ . Da mesma forma, após a ocorrência de  $t_2$ , a especificação *Underflow\_Ex<sub>2</sub>* ( $E_2$ ) inibe a ocorrência de  $Dex_2$  até que ocorra  $Lex_2$  ou  $t_{ok}$ .

A partir dos modelos apresentados, pode-se obter o modelo de especificação global, que é dado pela composição síncrona

$$E = \prod_{i=1}^{15} E_i$$

$E$  corresponde a um autômato com 28500 estados e 295000 transições, e pode ser combinado ao modelo da planta para fins de controle.



## 4 SÍNTESE DE CONTROLE: ANÁLISE ESTATÍSTICA

Este capítulo apresenta o resultado da composição entre os modelos do sistema quanto ao número de estados, número de transições e a ocorrência de maus estados, utilizando duas abordagens da TCS, a monolítica e a modular local.

### 4.1 ESTATÍSTICA DE CONTROLE MONOLÍTICO

Os modelos de  $G$  (planta) e  $E$  (especificação) foram obtidos por meio da modelagem de todos os componentes reais do sistema e a operação de composição síncrona entre cada um dos subsistemas envolvidos.

Uma vez que a planta  $G$  e a especificação  $E$  foram definidas, pode-se combinar os modelos para gerar o comportamento esperado sob controle, isto é, o comportamento em malha fechada, que é denotado por  $K$ , onde

$$K = E||G$$

Usando  $K$ , pode-se prosseguir com a síntese de controle, que é uma tarefa conduzida por meio da ferramenta *Supremica* (AKESSON et al., 2014).

$K$  é um autômato que modela a ação de controle  $E$  sobre  $G$ . Na prática, uma planta pode conter eventos não controláveis, involuntários e independentes de qualquer política de controle. Estes eventos podem levar o autômato  $K$  a possuir os chamados maus estados, cuja definição encontra-se na seção 2.3. Por este fato,  $K$  não é considerado o supervisor do sistema, porém, é utilizado para gerá-lo.

O procedimento consiste em identificar os possíveis estados ruins em  $K$ , que em geral são realmente encontrados e removidos, obtendo-se, assim, o autômato  $\text{sup}C(K, G)$ , que é o supervisor do sistema propriamente dito.

Simplificadamente, as etapas necessárias para se obter  $\text{sup}C(K, G)$  são:

- Identificar todos os componentes do sistema (sensores e atuadores) e fazer o levantamento dos eventos relacionados a cada um;
- Classificar cada evento como controlável ou não controlável;
- Utilizando uma ferramenta computacional (AKESSON et al., 2014; RUDIE et al., 2013; WONHAM, 2011; PINHEIRO et al., 2015; TORRICO, 2016), inserir os modelos de planta, especificações e os eventos em um novo projeto, fornecendo-os como parâmetros de entrada, graficamente ou numericamente, dependendo do software utilizado;
- Simular o comportamento do sistema, certificando-se de que os modelos atendem às necessidades de controle, se o software utilizado oferecer este recurso;
- Testar a controlabilidade do sistema e realizar a operação de composição síncrona entre os modelos de plantas (obtendo  $G$ ) e especificações (obtendo  $E$ ), por meio de recursos da ferramenta utilizada;
- Efetuar a composição síncrona entre os autômatos  $G$  e  $E$ , obtendo assim o autômato  $K$ ;
- Submeter  $K$  ao procedimento de sintetização, o qual realizará a remoção de possíveis maus estados, resultando no supervisor  $\text{sup}C(K, G)$ . Em caso de não existência de maus estados, o supervisor é igual a  $K$ .

Para a abordagem de modelagem apresentada,  $K$  é um modelo com 30697 estados e 119358 transições.

Por meio da ferramenta Supremica calcula-se  $\text{sup}C(K, G)$ . O resultado para este caso foi um modelo com 10195 estados e 36241 transições. Portanto,  $K$  apresenta 20502 maus estados. O processo de remoção dos maus estados de  $K$  é o que torna o modelo  $\text{sup}C(K, G)$  uma solução de controle ideal para o processo avícola.

Ressalta-se que, além das garantias formais de qualidade fornecidas pela TCS, a visão modular e de alto nível da abordagem também ajuda a evitar problemas de implementação caso o controlador fosse codificado manualmente, substituindo esse procedimento propenso a erros por um procedimento correto, modular e com garantia de recursos.

## 4.2 ESTATÍSTICA DE CONTROLE MODULAR LOCAL

Agora, uma nova versão do controlador  $\text{supC}(K, G)$  será obtida, dessa vez utilizando-se do CML, no intuito de obter o mesmo resultado, mas com economias computacionais. Esta abordagem oferece uma maneira de dividir as responsabilidades do controlador em vários controladores locais, que teoricamente são mais fáceis de serem implementados em *hardware*.

Utilizando o CML nesta aplicação para o cálculo de supervisores locais com a ferramenta Supremica, é necessário:

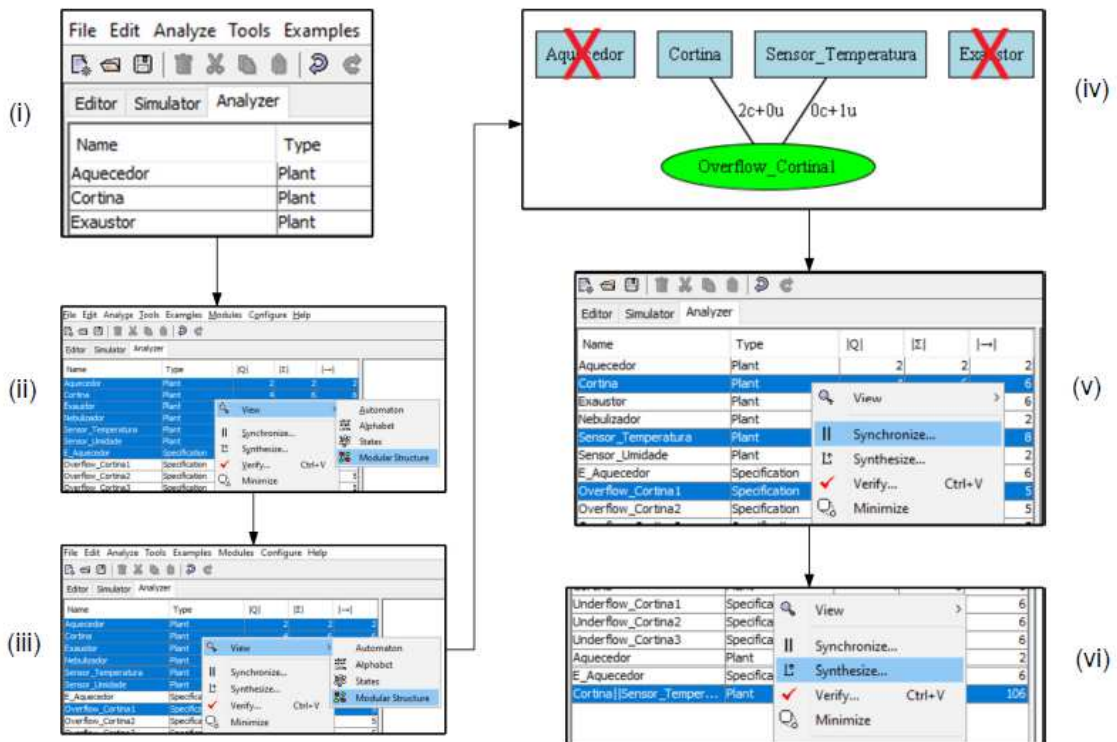
- (i) Acessar o ambiente *Analyzer* da ferramenta Supremica;
- (ii) Gerar uma estrutura modular envolvendo todos os subsistemas (plantas) existentes, mais uma especificação qualquer, e verificar a relação entre a especificação e os modelos da planta;
- (iii) Realizar a mesma operação para cada especificação do sistema;
- (iv) Em cada caso, descartar os modelos de planta que não possuem relação com a especificação correspondente;
- (v) Realizar a operação de composição síncrona entre cada especificação e os modelos com os quais se relacionam;
- (vi) Realizar a operação  $\text{supC}(K, G)$  sobre cada uma das estruturas modulares geradas. Estes serão os supervisores locais do sistema ( $S_{loc}$ ).

Os passos enumerados acima são exemplificados pela Figura 16.

A Figura 17 exemplifica, por meio do recurso gráfico do Supremica, o acoplamento entre modelos de subsistemas, o que leva à composição das plantas locais (item *Modular Structure* da ferramenta).

Tem-se que a especificação  $E_1$  possui relação direta com os modelos *Exaustor* e *Sensor\_Temperatura*, ou seja,  $G_{Ex}$  e  $G_{Temp}$ . Já a especificação  $E_9$  possui relação direta com os modelos *Cortina* e *Sensor\_Temperatura*, ou  $G_{Co}$  e  $G_{Temp}$ . Assim como nestas estruturas, para cada uma das 15 especificações da aplicação também é realizada a mesma operação e verificado com quais subsistemas cada especificação se relaciona.

Após o cálculo de todos os supervisores locais, tem-se a estatística apresentada na Tabela 1, em que:

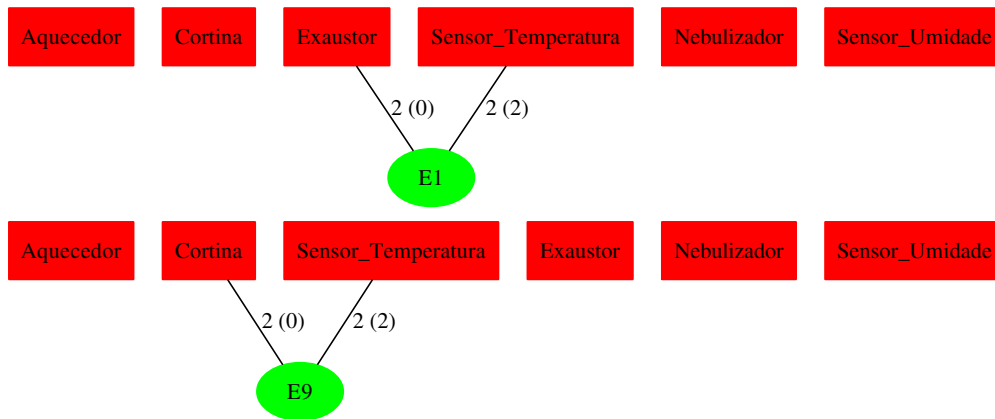


**Figura 16:** Exemplo de construção de supervisores locais na ferramenta Supremica

- $|Q|_K$ : Número de estados do autômato  $K$  local;
- $\rightarrow_K$ : Número de transições do autômato  $K$  local;
- $S_{loc}|Q|$ : Número de estados do supervisor local;
- $S_{loc} \rightarrow$ : Número de transições do supervisor local;
- $|\rightarrow \sum|$ : Número de eventos relacionados.

O modelo  $\text{sup}C_{loc}(K, G) = \|\|S_{loc}^i, i = 1, \dots, n$  denota o resultado da composição síncrona entre todos os supervisores locais obtidos, o que foi calculado para verificar que a abordagem CML resultou em supervisores que, atuando em conjunto, são equivalentes ao  $\text{sup}C(K, G)$ , possuindo o mesmo número de estados e de transições. Ou seja, computa-se o mesmo resultado, mas a partir de composições simples. A igualdade entre  $\text{sup}C_{loc}(K, G)$  e  $\text{sup}C(K, G)$  sugere a ausência de conflito entre os supervisores locais, o que efetivamente pode ser checado no Supremica.

Nota-se que cada supervisor local apresenta um número pequeno de estados quando comparados ao supervisor global do sistema. Em casos de implementação com limitações físicas de *hardware*, como por exemplo microcontroladores ou CLPs, este resultado implica em uma alternativa com menor dispêndio computacional (memória),



**Figura 17:** Exemplo de estruturas modulares

Especificação	Plantas Relacionadas	$ Q _K$	$\rightarrow_K$	$S_{loc} Q $	$S_{loc} \rightarrow$	$ \rightarrow \Sigma $
$E_1$	$G_{ex}, G_{temp}$	32	92	32	92	11
$E_2$	$G_{ex}, G_{temp}$	28	79	28	79	11
$E_3$	$G_{ex}, G_{temp}$	24	66	24	66	11
$E_4$	$G_{ex}, G_{temp}$	35	106	35	106	11
$E_5$	$G_{ex}, G_{temp}$	30	88	30	88	11
$E_6$	$G_{ex}, G_{temp}$	25	70	25	70	11
$E_7$	$G_{umid}, G_{temp}, G_{pad}$	36	109	36	109	9
$E_8$	$G_{umid}, G_{temp}, G_{pad}$	29	95	29	95	9
$E_9$	$G_{co}, G_{temp}$	32	92	32	92	11
$E_{10}$	$G_{co}, G_{temp}$	28	79	28	79	11
$E_{11}$	$G_{co}, G_{temp}$	24	66	24	66	11
$E_{12}$	$G_{co}, G_{temp}$	35	106	35	106	11
$E_{13}$	$G_{co}, G_{temp}$	30	88	30	88	11
$E_{14}$	$G_{co}, G_{temp}$	25	70	25	70	11
$E_{15}$	$G_{aq}, G_{temp}$	16	30	5	8	7
$\text{sup}C(K, G)$				10195	36241	23
$\text{sup}C_{loc}(K, G)$				10195	33241	23

**Tabela 1:** Estatística da composição dos supervisores locais.

uma vez que implementando o conjunto de supervisores locais, o comportamento geral do controlador permanece o mesmo (NOURELFATH; NIEL, 2004). Isso complementa as vantagens computacionais (processamento) da síntese modular em relação à centralizada.

## 5 ASPECTOS DE IMPLEMENTAÇÃO

Uma vez obtido o supervisor  $\text{sup}C(K, G)$  e também os supervisores locais  $S_{loc}^i$ , o próximo passo é decidir qual das duas abordagens apresenta melhores benefícios computacionais para a implementação do controlador.

O autômato do controlador é a base para a geração de código em formato de máquina de estados, para que possa ser implementado em *hardware* e ser finalmente integrado ao processo de controle prático, no contexto de um protótipo. Diferentes tipos de *hardware* podem ser utilizados para este fim, como os controladores lógicos programáveis (CLPs) e microcontroladores de baixo custo, por exemplo.

O controlador obtido pela abordagem CML apresenta menor custo computacional quando comparado ao controlador obtido pela abordagem monolítica.

### 5.1 GERAÇÃO DE CÓDIGO EM LINGUAGEM C

Como grande parte dos microcontroladores de baixo custo são geralmente programados em linguagem C, foram escolhidas ferramentas com base nessa exigência. Neste trabalho, a ferramenta *DESLAB* (TORRICO, 2016) foi utilizada para descrever o controlador na forma de um autômato, obtendo como resultado uma estrutura de código C (LORENCENA, 2016). O software, assim como a ferramenta *Supremica*, permite a representação de autômatos, recebendo como entrada a descrição das transições e estados de cada modelo individualmente, as operações de composição síncrona e checagem de propriedades.

Um diferencial do software *Deslab* é a geração de códigos em linguagem C, para microcontroladores, que implementam fisicamente o comportamento do sistema antes simulado. O código gerado é uma estrutura genérica de máquina de estado que representa o autômato supervisor do sistema,  $\text{sup}C(K, G)$ . O software permite que o usuário escolha entre a implementação do código em máquina de Mealy e máquina de

Moore, em linguagem C.

As máquinas de Mealy e Moore diferem-se uma da outra em questões de conceito e implementação, porém, seus comportamentos são idênticos. Na máquina de Mealy, cada estado do autômato contém apenas sua identificação (1, 2, 3...) e cada transição representa o valor das entradas e das saídas do sistema, ou seja, ao ocorrer um evento não controlável, o sistema executa uma ou mais ações na transição que o leva para o próximo estado. Já na máquina de Moore, cada estado contém sua identificação e também o valor das saídas do sistema correspondentes a ele, enquanto suas transições representam determinados eventos não controláveis a elas associados, executando então suas ações em cada estado, e não nas transições (MANTOVANI, 2010).

O software Deslab oferece a opção de código baseado em listas encadeadas e máquinas de estados com estruturas "case-if", dependendo do microcontrolador escolhido. O código exportado pelo Deslab pode ser utilizado para programar o microcontrolador que fará o gerenciamento das ações de controle, intermediando a comunicação do supervisor com os atuadores no meio físico.

Os dados dos autômatos que compõem o modelo do controlador devem ser fornecidos como parâmetros de entrada para o cálculo do supervisor e a geração de código. Cada evento controlável recebe um número ímpar, e cada evento não-controlável recebe um número par como identificador. Assim, os eventos são classificados como segue:

$$Lex_1(1), Lex_2(3), Lex_3(5), Dex_1(7), Dex_2(9), Dex_3(11), L_{pad}(13), D_{pad}(15), Lco_1(17), Lco_2(19), \\ Lco_3(21), Dco_1(23), Dco_2(25), Dco_3(27), L_{aq}(29), D_{aq}(31)$$

$$t_{ok}(2), t_1(4), t_2(6), t_3(8), t_{frio}(10), U_{ok}(12), U_{nok}(14)$$

Para a implementação do controlador na abordagem CML, cada uma das especificações deve ser combinada com as plantas às quais se relaciona, gerando os supervisores locais da mesma forma apresentada no software Supremica.

## 5.2 MONITORAMENTO DO SISTEMA SOB CONTROLE

O ambiente de implementação física do controlador limitou-se à simulação dos parâmetros de temperatura e umidade e à verificação das ações de controle no microcontrolador, pela indisponibilidade de um ambiente físico real para implantação do sistema de controle.

Implementado o controlador, inicia-se a fase de monitoramento do sistema sob controle. Essa fase é decisiva em termos práticos, pois implica diretamente em custos financeiros.

As operações que compõem um sistema de controle para processos avícolas geralmente não tratam de equipamentos isolados, mas sim de um conjunto de ações baseadas em uma série de fatores (como temperatura e umidade) que envolvem um ou mais equipamentos. O especialista atenta para o perfeito funcionamento destes, que, uma vez defeituosos, podem causar grandes prejuízos financeiros por mortalidade (LAGANÁ, 2005; SILVA; VIEIRA, 2010).

Não são raros relatos de perdas de lotes inteiros de frango devido a problemas momentâneos de controle. Uma simples interrupção do sistema de exaustão, por exemplo, por alguns minutos, seria suficiente para comprometer um lote. Ademais, desde a chegada das aves no aviário até o dia em que são entregues para o abate, é necessário o monitoramento empírico das variáveis ambientes de maneira a se certificar de que elas estejam ideais para a respectiva fase de desenvolvimento dos frangos. Por este motivo, a presença do avicultor é indispensável, o que caracteriza o manejo avícola uma atividade exaustiva e minuciosa. Um recurso que se encaixa naturalmente às necessidades do manejo avícola junto ao controlador é o uso de sistemas de aquisição de dados e monitoramento remoto.

Em entrevistas feitas com avicultores da região, foram relatados episódios em que os responsáveis pelos aviários tiveram que deslocar-se aos galpões várias vezes durante o dia e a noite, ou permanecerem próximos à propriedade, sem poderem se ausentar. Isso se deve ao fato de que, mesmo utilizando sistemas automáticos de controle de temperatura e de umidade, os parâmetros do processo devem ser alimentados manualmente e acompanhados constantemente. Os avicultores argumentam ainda sobre a necessidade de estarem presentes no aviário para que, com sua experiência, analisem o comportamento das aves e, empiricamente, verifiquem o seu bem-estar.

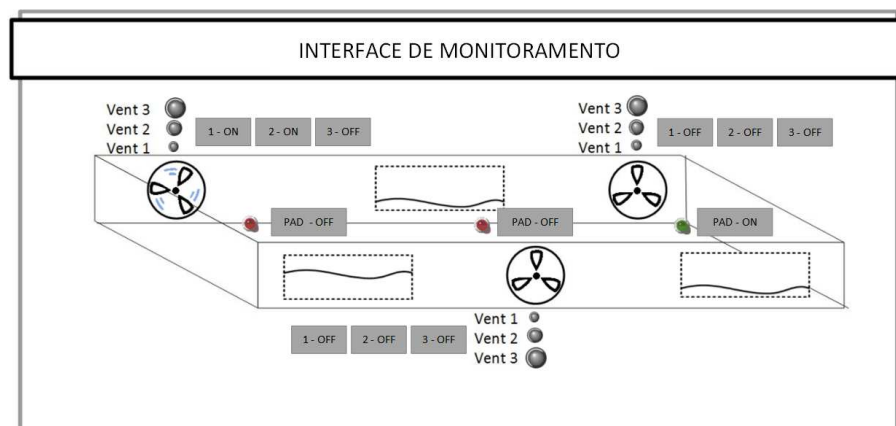
Alguns trabalhos, inclusive, investigam a análise de imagens e sons para deter-



minar o bem-estar das aves, que é visivelmente distinto para cada parâmetro climático (PEREIRA et al., 2013; NAAS, 2011; SILVA, 2007; MOGAMI, 2009) resultando em maior movimentação, aglomeração, retração, etc.

Através de um sistema de controle e monitoramento remoto, é possível realizar o monitoramento por meio de um computador ou *smartphone* conectado à internet ou a uma rede local. O ScadaBr é um recurso computacional da família dos sistemas SCADA (*Supervisory Control and Data Acquisition*), nacional e de código aberto, que permite desenvolver esta tarefa. Através deste *framework*, o comportamento dos frangos pode ser acompanhado de diversas maneiras, como pela transmissão da imagem de câmeras instaladas no interior dos galpões, pela coleta dos parâmetros de temperatura e umidade via sensores, entre outros. Além disso, o ScadaBR conta com uma estrutura de alarmes que alertam para eventuais defeitos em equipamentos ou mau funcionamento do sistema. Há ainda a possibilidade de atuação via interface remota. Em conjunção, esses aspectos tornam mais eventual a presença do avicultor no galpão.

No escopo deste trabalho, o ScadaBR é utilizado para monitorar o processo avícola por meio da comunicação entre um microcontrolador e os dispositivos de atuação dispostos no aviário, em um ambiente computacional (LORENCENA, 2016). Com base nessa comunicação, é implementada uma interface que apresenta ao usuário os valores ou níveis medidos pelos sensores nos pontos de aquisição de dados e a ação em tempo real dos equipamentos de atuação. Além disso, a interface gráfica provê a interferência manual do especialista sobre o processo automático.



**Figura 18:** Interface gráfica utilizando componentes do software ScadaBR

## 6 CONCLUSÃO

Este trabalho propôs uma abordagem para melhorar e facilitar o manejo avícola, fornecendo um modelo computacional para gerenciar o controle das variáveis de ambiência e o monitoramento das condições de vida das aves. A proposta é baseada em técnicas de teoria de controle formal, integradas a uma aplicação prática com o suporte de ferramentas computacionais.

Para viabilizar esta abordagem, uma arquitetura de aviário foi adotada, a fim de fornecer mais detalhes sobre suas condições climáticas internas. Destaca-se que esta solução de controle pode ser aplicada a qualquer arquitetura de aviários, e que, além dos galpões de criação de frangos de corte, também é possível sua adaptação para aviários de galinhas poedeiras, suinocultura e demais atividades de criação animal que exijam ambiente controlado, e ainda em cultura de vegetais.

É importante ressaltar também que esta abordagem tem potencial para proporcionar benefícios ecológicos a um processo avícola: à medida que o controle climático se torna mais modular, a atuação torna-se mais local, o que implica que o equilíbrio térmico poderia ser alcançado com menos consumo de energia. Considera-se essa validação além dos limites deste trabalho, mas inclui-se como uma perspectiva de pesquisa futura.

Em resumo, as principais vantagens do modelo proposto podem ser apresentadas da seguinte forma:

- A leitura das condições ambientais é mais ampla do que os métodos atuais, o que aumenta as possibilidades de atuação em diferentes locais;
- O controle é comprovadamente não bloqueante e minimamente restritivo;
- A conformidade com o conjunto de especificações do sistema é formalmente garantida pela TCS;

- O controlador pode ser modelado e programado através de ferramentas computacionais e implementado através da geração automática de código;
- O controle incorpora o fato de que os eventos podem ser dependentes dentro de um arranjo físico;
- A escalabilidade do sistema é melhorada, pois cada setor de ventilação é controlado individualmente;
- O monitoramento e o controle de variáveis podem ser realizados remotamente;
- A confiabilidade e a produtividade são aumentadas em relação ao manejo avícola convencional, pois as ações humanas são naturalmente propensas a erros, ao mesmo tempo em que assegura-se que o sistema seguirá um padrão pré-definido;
- O modelo proposto para aviários em conjunto com o controlador modular promove um ambiente mais homogêneo em relação às condições climáticas;
- A presença constante do especialista humano no aviário é substituída por mecanismos de intervenção remota e alarmes;
- A abordagem tem potencial para ser eficiente em termos energéticos.

Perspectivas futuras visam integrar o sistema de controle proposto a uma estrutura de decisão inteligente capaz de sugerir ações a serem tomadas, com base em trabalhos já realizados nesta linha, que envolvem otimização e planejamento do manejo avícola (RIBEIRO et al., 2015).

A integração entre as tecnologias citadas reduziria ainda mais a interferência de especialistas humanos no processo, substituindo suas decisões por ações mais confiáveis, testadas computacionalmente. O sistema de decisão sugere ações que precisam ser executadas por um controlador capaz de absorver as informações providas pelo otimizador e transformá-las em reações.

A aplicação dos formalismos da TCS em processos avícolas é uma abordagem que teve início no projeto de Iniciação Científica intitulado "*Avicultura Inteligente: Técnicas de Inteligência Artificial para Melhorar Processos Avícolas*", resultando no trabalho *Um Sistema de Controle para Processos Avícolas com Fluxo de Ar Transversal*. A pesquisa foi apoiada pelo CNPq, sobre o processo 484859/2013-7, como apoio da Universidade Tecnológica Federal do Paraná - UTFPR, por meio do programa de bolsas de iniciação científica - PIBIC 2014/2015, CNPq-Af.

A continuidade da pesquisa resultou na publicação de um artigo intitulado *Contribuições ao Manejo Avícola Utilizando Técnicas de Controle Supervisório*, apresentado no Congresso INDUSCON 2016 - *12th IEEE/IAS International Conference on Industry Applications*, na cidade de Curitiba - PR. (LORENCENA et al., 2016).

A proposta dissertada foi aprovada em formato de projeto de pesquisa com auxílio financeiro e custeio de bolsas, sob o Número do Processo 402145/2016-0, e Chamada Universal 2016, com o título: *Contribuições ao manejo avícola usando técnicas de controle supervisório e de monitoramento remoto*.

## REFERÊNCIAS

- ABPA. **A Avicultura Brasileira**. 2015. EMBRAPA. Disponível em: <<http://www.ubabef.com.br>>.
- AKESSON, K. et al. **Supremica**. [S.l.], 2014. Disponível em: <<http://www.supremica.org/>>.
- AMARAL, A. et al. Efeito do ambiente de produção sobre frangos de corte sexados criados em galpão comercial. **Arquivo Brasileiro de Medicina Veterinária e Zootecnia**, v. 63, p. 649 – 658, 2011.
- AVIAGEN. **Broiler Management Handbook**. [S.l.], 2014. Disponível em: <[www.aviagen.com](http://www.aviagen.com)>.
- AYDIN, A. Using 3d vision camera system to automatically assess the level of inactivity in broiler chickens. **Computers and Electronics in Agriculture**, v. 135, p. 4 – 10, 2017. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169917301400>>.
- BICHARA, T. **Aviário azul e dark house para frangos de corte - Desenvolvendo novos conceitos para aviários de pressão negativa**. 2011. NFT ALLIANCE.
- BLANES-VIDAL, V.; FITAS, V.; TORRES, A. Differential pressure as a control parameter for ventilation in poultry houses: Effect on air velocity in the zone occupied by animals. v. 5, p. 31–37, 03 2007.
- BLANES-VIDAL, V. et al. Application of computational fluid dynamics to the prediction of airflow in a mechanically ventilated commercial poultry building. **Biosystems Engineering**, v. 100, n. 1, p. 105 – 116, 2008. ISSN 1537-5110. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S153751100800055X>>.
- BREWER, R. N.; JR., C. A. F.; KOON, J. L. Heating poultry houses. **Poultry Science**, v. 60, n. 5, p. 917, 1981.
- BRIGANO, M. **Pontos Críticos no Manejo e Ambiência de Verão**. 2013. Ave World. Disponível em: <<http://www.aveworld.com.br/noticia/pontos-criticos-no-manejo-e-ambiencia-de-verao>>.
- BRUZUAL, J. J. et al. Effects of relative humidity during the last five days of incubation and brooding temperature on performance of broiler chicks from young broiler breeders. **Poultry Science**, v. 79, p. 1385–1391, 2000.
- BUSTAMANTE, E. et al. Exploring ventilation efficiency in poultry buildings: The validation of computational fluid dynamics (cfd) in a cross-mechanically ventilated broiler farm. **Energies**, v. 6, n. 4, p. 535–542, 1994.

CANDIDO, M. G. L. et al. Determination of thermal comfort zone for early-stage broilers. **Engenharia Agrícola**, scielo, v. 36, p. 760 – 767, 10 2016. ISSN 0100-6916.

CANGAR, . et al. Quantification of the spatial distribution of surface temperatures of broilers. **Poultry Science**, v. 87, n. 12, p. 2493, 2008.

CASSANDRAS, C. G.; LAFORTUNE, S. **Introduction to Discrete Event Systems**. 2. ed. NY: Springer Science, 2008.

CASSUCE, D. C. et al. Thermotolerance Acquisition in Broiler Chickens by Temperature Conditioning Early in Life: The Effect of Timing and Ambient Temperature. **Poultry Science**, v. 80, p. 1662–1666, 12 2001.

CASSUCE, D. C. et al. Thermal comfort temperature update for broiler chickens up to 21 days of age. **Engenharia Agrícola**, scielo, v. 33, p. 28 – 36, 02 2013. ISSN 0100-6916.

CATELAN, F. **Sistemas de Aquecimento Automáticos na Produção de Frangos de Corte**. Tese (Doutorado) — Universidade Estadual do Oeste do Paraná, Cascavel - Paraná, 2007.

CIAS. **Resultado do ICP Frango/Embrapa de Abril de 2016**. 2016. EMBRAPA. Disponível em: <<http://www.cnpsa.embrapa.br/cias/>>.

CRUZ, C. R. da. Bem-estar animal no cenário internacional. **IV Simpósio Brasil Sul de Avicultura**, Chapecó, Santa Catarina, Brasil, Abril 2003.

CURY, J. E. R. Teoria de controle supervisorio de sistemas a eventos discretos. **V Simpósio Brasileiro de Automação Inteligente**, 2001.

CURY, J. E. R. et al. Supervisory control of discrete event systems with distinguishers. **Automatica**, v. 56, p. 93–104, 2015.

DAWKINS, M. S.; DONNELLY, C. A.; JONES, T. A. Chicken welfare is influenced more by housing conditions than by stocking density. **Nature**, v. 427, 2004.

DONKOH, A. Ambient temperature: a factor affecting performance and physiological response of broiler chickens. **International Journal of Biometeorology**, v. 33, n. 4, p. 259–265, 1989.

DONKOH, A.; ATUAHENE, C. Management of environmental temperature and rations for poultry production in the hot and humid tropics. **International Journal of Biometeorology**, v. 32, p. 247–253, 1988.

DOWELL, R. M. Bases biológicas de la producción animal en zonas tropicales. **Zaragoza**, 1974.

FILHO, C. S. **Protocolos Orientados a Caracter**. 2002. Disponível em: <<http://www.cpdee.ufmg.br/seixas/PaginaSDA/Download/DownloadFiles/ProtocolosCaracter.PDF>>.

FUNCK, S. R.; FONSECA, R. A. Avaliação energética e de desempenho de frangos com aquecimento automático a gás e a lenha. **Revista Brasileira de Engenharia Agrícola e Ambiental**, v. 12, n. 1, p. 91–97, Setembro 2007.

GAO, Y.; RAMIREZ, B. C.; HOFF, S. J. Omnidirectional thermal anemometer for low airspeed and multi-point measurement applications. **Computers and Electronics in Agriculture**, v. 127, p. 439 – 450, 2016.

GATES, R. S.; XIN, H. Extracting poultry behaviour from time-series weigh scale records. **Computers and Electronics in Agriculture**, v. 62, n. 1, p. 8 – 14, 2008. Precision Livestock Farming (PLF).

HARPER, L. A.; FLESCHE, T. K.; WILSON, J. D. Ammonia emissions from broiler production in the san joaquin valley. **Poultry Science**, n. 89, p. 1802–1814, 2010.

HULZEBOSCH, J. Effective heating systems for poultry houses. **World Poultry**, v. 22, n. 2, 2006.

KREVIINGHAUS, R. E. et al. Method and apparatus for controlling temperature within poultry houses and the like. US, n. 5,492,082, 02 1996. Patent. Filed: Jan. 18,1994. Filing Num.: 183,125. Disponível em: <<https://goo.gl/UBab7v>>.

LAGANÁ, C. **Otimização da Produção de Frango de Corte em Condições de Estresse por Calor**. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, Porto Alegre - RS, Janeiro 2005.

LIN, H. et al. Thermoregulation responses of broiler chickens to humidity at different ambient temperatures. i. one week of age. **Poultry Science**, n. 84, p. 1166–1172, 2005.

LIN, T. et al. Development of {MOS} sensor-based {NH<sub>3</sub>} monitor for use in poultry houses. **Computers and Electronics in Agriculture**, v. 127, p. 708 – 715, 2016. ISSN 0168-1699.

LOPES, A. Z. **Contribuições Para o Desenvolvimento Sustentável de Um Sistema de Monitoramento e Controle de Galpões para Frangos de Corte**. Tese (Doutorado) — UNICAMP, Campinas, Abril 2015. Tese de Doutorado - Faculdade de Engenharia Elétrica e de Computação - Departamento de Semicondutores e Instrumentos.

LORENCENA, M. C. **Poultry Farming Process Controller**. [S.l.], 2016. Disponível em: <<https://goo.gl/XMh4te>>.

LORENCENA, M. C. et al. **Contribuições ao Manejo Avícola Utilizando Técnicas de Controle Supervisório**. 2016. 12th IEEE/IAS International Conference on Industry Applications - INDUSCON.

MANTOVANI, S. C. A. **Circuitos Sequenciais (Máquinas Síncronas ou de Estados Finitos)**. 2010.

MENEGALI, I. **Diagnóstico da Qualidade do Ar na Produção de Frangos de Corte em Instalações Semi-Climatizadas Por Pressão Negativa e Positiva, no Inverno, no Sul do Brasil**. Tese (Doutorado) — Universidade Federal de Viçosa, Viçosa, Minas Gerais, Brasil, 2005.

MIRZAEI-GHALEH, E. et al. Comparison of fuzzy and on/off controllers for winter season indoor climate management in a model poultry house. **Computers and Electronics in Agriculture**, v. 110, p. 187 – 195, 2015.

- MODICON. **Modicon Modbus Protocol Reference Guide**. 1996. MODICON, Inc., Industrial Automation Systems.
- MOGAMI, C. A. **Desenvolvimento de Metodologias para Determinação do Bem-Estar e Massa Corporal de Frangos de Corte Por Meio de Análise Digital de Imagens**. Dissertação (Mestrado) — Universidade Federal de Viçosa, Viçosa - MG, 2009.
- MOOR, T. A discussion of fault-tolerant supervisory control in terms of formal languages. **International Federation of Automatic Control**, 2016.
- MORTENSEN, A. K.; LISOUSKI, P.; AHRENDT, P. Weight prediction of broiler chickens using 3d computer vision. **Computers and Electronics in Agriculture**, v. 123, p. 319 – 326, 2016.
- NAAS, I. de A. Uso de técnicas de precisão na produção animal. **Revista Brasileira de Zootecnia**, v. 40, p. 358–364, 2011.
- NASCIMENTO, G. et al. Assessment of broiler surface temperature variation when exposed to different air temperatures. **Revista Brasileira de Ciência Avícola**, scielo, v. 13, p. 259 – 263, 12 2011. ISSN 1516-635X.
- NASCIMENTO, G. A. J. do et al. Efeitos da substituição do milho pela raspa de mandioca na alimentação de frangos de corte, durante as fases de engorda e final. **Ciência e Agrotecnologia**, 2005.
- NOBLE, F. D. O.; TEETER, R. G. Components of feed efficiency in broiler breeding stock: The use of fasted body temperature as an indicator trait for feed conversion in broiler chickens. **Poultry Science**, v. 83, n. 4, p. 515–520, 2004.
- NOURELFATH, M.; NIEL, E. Modular supervisory control of an experimental automated manufacturing system. **Control Engineering Practice**, v. 12, n. 2, p. 205 – 216, 2004.
- OGATA, K. **Modern Control Engineering**. [S.l.]: Prentice Hall, 2003.
- PEREIRA, D.; NÄÄS, I. Estimating the thermoneutral zone for broiler breeders using behavioral analysis. **Computers and Electronics in Agriculture**, v. 62, n. 1, p. 2 – 7, 2008. ISSN 0168-1699. Precision Livestock Farming (PLF).
- PEREIRA, D. F. et al. Machine vision to identify broiler breeder behavior. **Computers and Electronics in Agriculture**, 2013.
- PINHEIRO, L. P. et al. Nadzoru: A software tool for supervisory control of discrete event systems. 2015.
- PRINCE, R. P. et al. Response of chickens to temperature and relative humidity environments<sup>1</sup>. **Poultry Science**, v. 44, n. 1, p. 73, 1965.
- QUEIROZ, M. H. D.; CURY, J. E. R. Modular supervisory control of large scale discrete event systems. In: **Discrete Event Systems: Analysis and Control. Proc. WODES'00**. [S.l.]: Kluwer Academic, 2000. p. 103–110.



- RAMADGE, P. J.; WONHAM, W. M. Modular feedback logic for discrete event systems. **SIAM Journal of Control and Optimization**, v. 25, n. 5, p. 1202–1218, 1987.
- RAMADGE, P. J.; WONHAM, W. M. Supervisory control of a class of discrete event process. **SIAM Journal of Control and Optimization**, v. 25, n. 1, p. 206–230, 1987.
- RAMADGE, P. J.; WONHAM, W. M. The control of discrete event systems. **Discrete Event Dynamic Systems**, v. 77, p. 81–98, 1989.
- REECE, F. N.; DEATON, J. W.; KUBENA, L. F. Effects of high temperature and humidity on heat prostration of broiler chickens. **Poultry Science**, v. 51, n. 6, p. 2021, 1972.
- RIBEIRO, R. et al. A learning model for intelligent agents applied to poultry farming. In: **INSTICC. Proceedings of the 17th International Conference on Enterprise Information Systems - Volume 1: ICEIS**. [S.l.]: ScitePress, 2015. p. 495–503. ISBN 978-989-758-096-3.
- RONCHI, C. **Principais Práticas de Manejo Para Aves Recém-Nascidas**. [S.l.], Dez/Jan 2004.
- RUDIE, K. et al. **Ides - Integrated Discrete-Event Systems**. [S.l.], 2013. Disponível em: <<https://qshare.queensu.ca/Users01/rudie/www/software.html>>.
- SALLABI, F. et al. Design and implementation of an electronic mobile poultry production documentation system. **Computers and Electronics in Agriculture**, v. 76, n. 1, p. 28 – 37, 2011.
- SHALLIT, J. O.; SHALLIT, J. **A second course in formal languages and automata theory**. [S.l.]: Cambridge University Press Cambridge, 2009.
- SILVA, I.; VIEIRA, F. Ambiência animal e as perdas produtivas no manejo pré-abate: O caso da avicultura de corte brasileira. **Archivos de Zootecnia**, n. 59, p. 113–131, Setembro 2010.
- SILVA, I. J. O. da. **Contribuições à Zootecnia de Precisão na Produção Industrial de Aves e Suínos no Brasil**. Piracicaba - SP: Universidade de São Paulo, 2007.
- SO-IN, C.; POOLSANGUAN, S.; RUJIRAKUL, K. A hybrid mobile environmental and population density management system for smart poultry farms. **Computers and Electronics in Agriculture**, v. 109, p. 287 – 301, 2014.
- SOLDATOS, A. et al. Nonlinear robust temperature-humidity control in livestock buildings. **Computers and Electronics in Agriculture**, v. 49, n. 3, p. 357 – 376, 2005.
- SOUZA, V. L. F. de et al. Efeito da automatização nas diferentes estações do ano sobre os parâmetros de desempenho, rendimento e qualidade da carne de frangos de corte. **Acta Scientiarum. Animal Sciences**, 2010.
- TEIXEIRA, M. et al. Supervisory control of des with extended finite-state machines and variable abstraction. **Automatic Control, IEEE Transactions on**, v. 60, n. 1, p. 118–129, 2014.
- TORRICO, C. **Deslab 3.6**. [S.l.], 2016. Disponível em: <<https://goo.gl/WNbwJz>>.

- UBABEF. **Protocolo de Bem-Estar para Frangos e Perus**. 2008. União Brasileira de Avicultura (UBA).
- VANTRESS, C. **Manual de Manejo de Frangos de Corte**. [S.l.], 2008. Disponível em: <<http://wp.ufpel.edu.br/avicultura/files/2012/04/Cobb-Manual-Frango-Corte-BR.pdf>>.
- VIEIRA, F. M. C. et al. Preslaughter mortality of broilers in relation to lairage and season in a subtropical climate. **Poultry Science**, v. 90, n. 10, p. 2127, 2011.
- VIEIRA, N. P. **Conceitos de Aviários 'Pressão Negativa' - Circular Técnica**. 2009. Aviagen Brasil.
- WEG, E. E. **Manual do Usuário Modbus RTU**. 2013. Disponível em: <<http://ecatalog.weg.net>>.
- WONHAM, W. M. **TCT - Design Software**. [S.l.], 2011. Disponível em: <<http://www.control.utoronto.ca/~wonham>>.
- XIN, H.; BERRY, I.; COSTELLO, T. A computerized measurement and data acquisition system for field poultry research. **Computers and Electronics in Agriculture**, v. 11, n. 2, p. 143 – 156, 1994.
- XIN, H. et al. Temperature and humidity profiles of broiler houses with experimental conventional and tunnel ventilation systems. **Agricultural and Biosystems Engineering**, v. 10, n. 5, p. 2605–2623, 2013.
- ZHU, S.; LEE, S. Co-combustion performance of poultry wastes and natural gas in the advanced swirling fluidized bed combustor (sfbc). **Waste Management**, v. 25, n. 5, p. 511 – 518, 2005.

## APÊNDICE A - SIMULAÇÃO DOS PARÂMETROS DE AMBIÊNCIA

Por meio de visitas a aviários e pesquisas pelos padrões das principais empresas mediadoras de frangos de corte do Brasil, percebe-se que a grande maioria dos galpões apresenta as particularidades citadas pela literatura e possuem os mesmos desafios relacionados à manutenção da ambiência adequada para o desenvolvimento animal, mesmo os que dispõem de tecnologia. Tais observações não são publicadas pelas empresas integradoras.

É notável que, devido à complexidade existente no processo de criação de frangos de corte e a sensibilidade das aves em relação às condições térmicas ambientais, é imprescindível que haja algum mecanismo capaz de controlar todos os atuadores presentes em um aviário, sejam eles aquecedores, resfriadores ou umidificadores, a fim de automatizar o processo, dispensando as tomadas de decisões empíricas e suscetíveis a erros realizadas pelo avicultor.

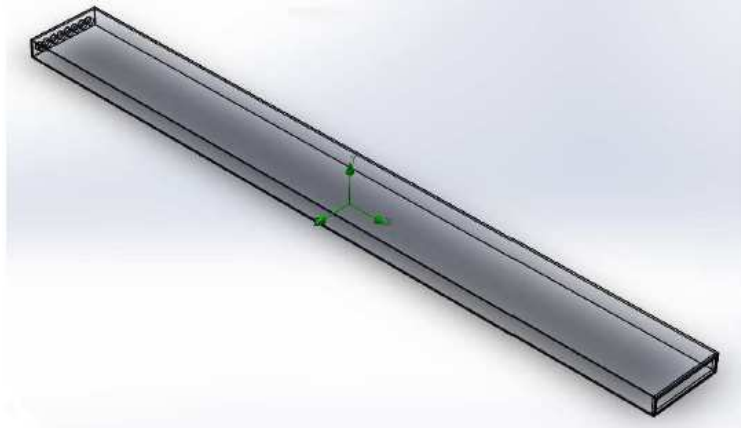
É necessário ainda que estas ações de controle sejam capazes de promover o máximo de homogeneização possível no ambiente. Neste trabalho, a escolha da arquitetura com fluxo de ar transversal está relacionada a este fato.

Além de apresentar melhoras na homogeneização em relação à arquitetura com fluxo de ar longitudinal, a arquitetura de fluxo transversal permite que o sistema de controle realize ações mais pontuais, ou seja, através dos setores de ventilação, é possível efetuar a leitura de temperatura e umidade em um número maior de pontos ao longo do aviário, e acionar os atuadores pontualmente, sendo que cada um deles atinge uma zona menor, por possuir um fluxo de influência transversal, e não longitudinal.

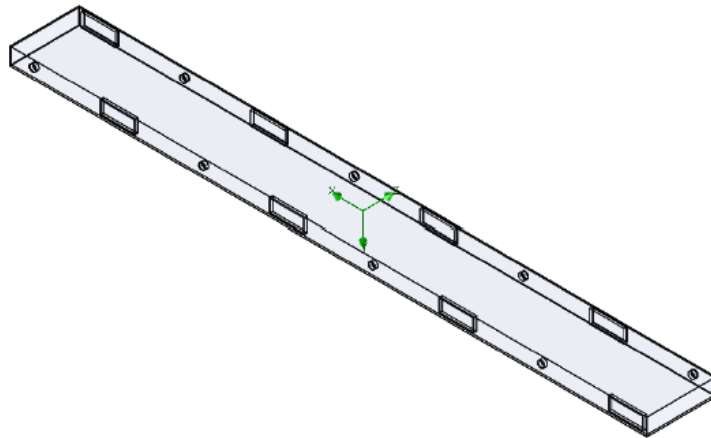
A fim de verificar o comportamento das faixas de temperatura e umidade em cada uma das arquiteturas para aviários apresentadas, uma simulação foi realizada utilizando o *software SolidWorks*, por meio da ferramenta *Flow Simulation*.

Foram construídos dois cenários que representam ambientes físicos reais com a arquitetura tradicional, como mostra a Figura 19, e a arquitetura proposta, na Figura

20.



**Figura 19:** Modelo tradicional de aviário para simulação no *SolidWorks*



**Figura 20:** Modelo proposto de aviário para simulação no *SolidWorks*

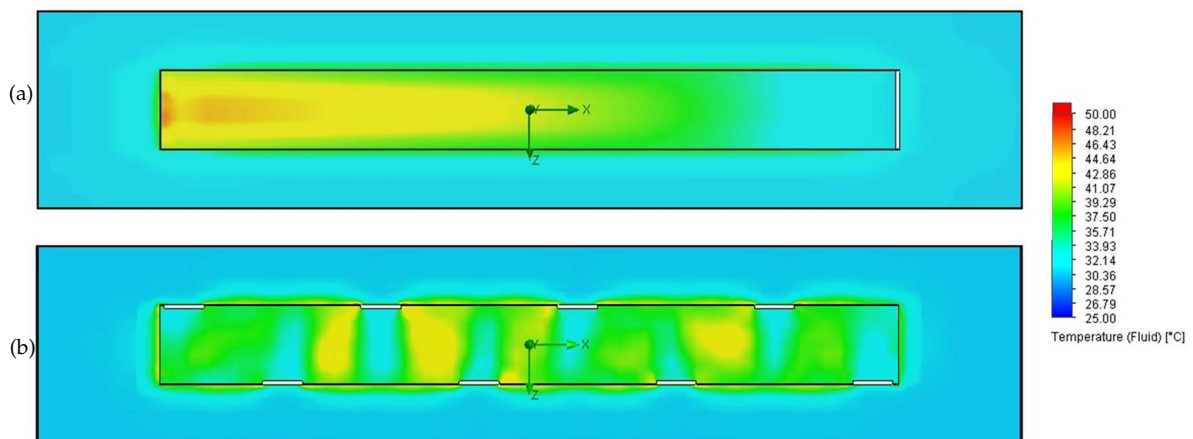
O aviário simulado possui 8 setores de ventilação, como mostra a Figura 20. O modelo tradicional da Figura 19 é composto por exaustores, cortinas, nebulizadores e aquecedores. Já o modelo proposto é composto por exaustores, cortinas, *pad-coolings* e aquecedores. A situação real simulada exige uma ação de resfriamento, portanto, nesse caso, o aquecedor não está sendo considerado.

São definidas as seguintes condições para as simulações em ambas as arquiteturas:

- Dimensões do aviário (largura  $\times$  comprimento  $\times$  altura ):  $16m \times 150m \times 4m$ ;
- Quantidade de exaustores: 8;
- Dimensões das cortinas:  $0,5m \times 3m$ ;

- Vazão dos exaustores:  $2m^3/s$ ;
- Diâmetro dos exaustores:  $1,5m$ ;
- Temperatura externa:  $30^{\circ}C$ ;
- Umidade relativa: 50%;
- Condição do tempo: Ensolarado;
- Horário: 12h00min.

A Figura 21 apresenta um comparativo do comportamento das faixas de temperatura nas duas arquiteturas.



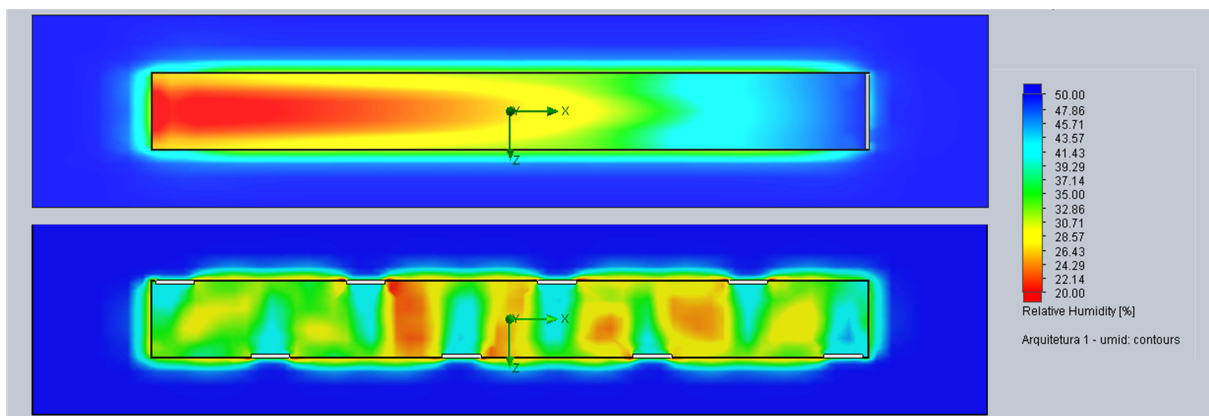
**Figura 21:** Distribuição das faixas internas de temperatura via simulação na arquitetura com fluxo de ar longitudinal (quadro superior) e na arquitetura como fluxo de ar transversal (quadro inferior)

Por meio desta simulação é possível identificar a distribuição das faixas de temperatura no interior dos ambientes. Na primeira arquitetura, Figura 21(a), é visível que a temperatura é bastante heterogênea, causando uma sensação de frio para os frangos que localizam-se nas proximidades da entrada de ar e sensação de calor para aqueles que estão próximos à saída de ar. Esta discrepância entre as extremidades é gerada devido ao fluxo de ar longitudinal, em que o ar adentra no aviário com a temperatura externa e, ao percorrer todo o ambiente aumenta sua própria temperatura. Esta diferença é perceptível aos humanos, e foi constatada em todos os aviários visitados na região que possuem esta configuração.

Na segunda arquitetura, que foi a escolhida como base para o sistema de controle, mostrada na Figura 21(b), a distribuição das faixas de temperatura torna-se mais homogênea devido ao fluxo de ar transversal que promove maior circulação interna, reduzindo a discrepância dos parâmetros medidos em diferentes pontos do ambiente. Nesta configuração, as ações de controle atingem áreas mais pontuais e distribuídas.

O controle de umidade também apresenta um comportamento similar, como mostra a Figura 22, uma vez que, na arquitetura com fluxo de ar longitudinal, são considerados apenas dois pontos de nebulização, e à medida que o ar puro adentra no ambiente refrescando-o, a ação intensa dos exaustores ocasiona considerável perda de umidade em suas proximidades.

Na arquitetura com fluxo de ar transversal, os nebulizadores são substituídos por *pad-coolings* que liberam vapor auxiliando no arrefecimento do ambiente, sem alterar bruscamente os índices de umidade relativa, mantendo o ambiente visivelmente mais homogêneo.



**Figura 22:** Distribuição das faixas internas de umidade via simulação na arquitetura com fluxo de ar longitudinal (quadro superior) e na arquitetura com fluxo de ar transversal (quadro inferior)

## APÊNDICE B - PROTOCOLO MODBUS RTU

O protocolo Modbus foi desenvolvido pela *Modicon Industrial Automation Systems*, a atual *Schneider Electric*, para servir como um meio de comunicação entre dispositivos industriais, sendo eles *um mestre* e *um ou mais escravos*. Embora seja utilizado normalmente sobre conexões seriais padrão *RS – 232* e *RS – 485*, ele também pode ser utilizado como um protocolo da camada de aplicação de redes industriais, tais como *TCP/IP* sobre *Ethernet* e *MAP* (FILHO, 2002).

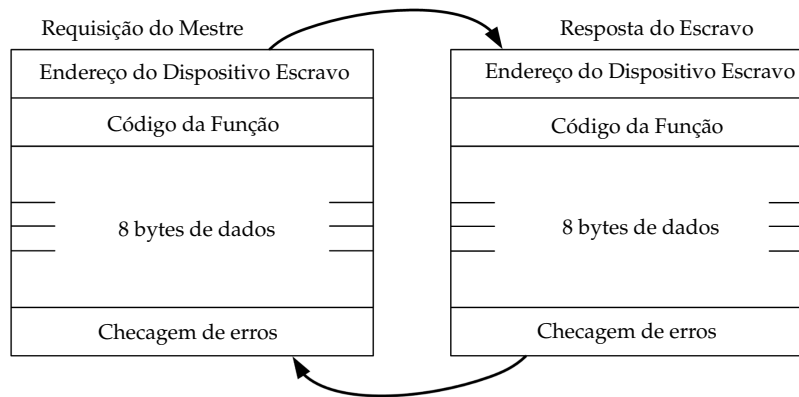
Este protocolo define uma estrutura de mensagens reconhecidas por dispositivos independentemente do tipo de rede sobre a qual se comunicam. Ele descreve a maneira como um controlador realiza pedidos e responde outros elementos do sistema, segundo alguma função pré-definida, além de como os erros de envio e recepção de mensagens são detectados e notificados, estabelecendo um formato comum para o *layout* e conteúdo dos campos das mensagens (MODICON, 1996).

Segundo a documentação fornecida pela *Modicon*, os controladores que utilizam este protocolo de comunicação, em redes *Modbus* padrão, podem ser configurados para se comunicar utilizando um dos dois modos de transmissão existentes: *ASCII* e *RTU*. Um desenvolvedor pode implementar qualquer um dos dois modos, considerando os parâmetros relevantes da porta de comunicação serial a ser utilizada, como: identificação da porta serial, taxa de transmissão, modo de paridade, etc. O modo de transmissão e os parâmetros de configuração serial devem ser os mesmos para todos os dispositivos presentes na rede *Modbus*.

O chamado protocolo *Modbus* padrão, no qual existem os modos *ASCII* e *RTU*, define o conteúdo dos campos de mensagens, que são arranjos de bits transmitidos por meio de uma comunicação serial, e como as informações são organizadas nos campos do pacote de dados, além de como serão decodificadas pelo dispositivo que as receberem.

A comunicação entre dispositivos utilizando o protocolo *Modbus* é baseada no modelo mestre-escravo, onde um único dispositivo da rede, o mestre, pode iniciar

transações, ou troca de mensagens, e os demais, os escravos, respondem com o fornecimento de dados requisitados pelo mestre, ou executando uma ação por ele solicitada (FILHO, 2002). Quando o dispositivo mestre envia uma mensagem endereçada a um escravo, apenas o dispositivo endereçado retorna uma resposta, como mostra a Figura 23.



**Figura 23:** Ciclo de mensagens genéricas do Modbus trocadas entre mestre e escravo

O mestre indica o início de uma comunicação enviando um byte com o endereço do escravo para o qual se destina a mensagem. Ao enviar a resposta, o escravo também inicia o telegrama com o seu próprio endereço. O campo *código da função* também contém um único byte, onde o mestre especifica o tipo de serviço ou função solicitada ao escravo (leitura, escrita, entre outras). De acordo com o protocolo, cada função é utilizada para acessar um tipo específico de dado. As funções são pré-definidas para este protocolo. O campo *bytes de dados* possui tamanho variável e o formato e conteúdo deste campo dependem da função utilizada e dos valores transmitidos (WEG, 2013). Por último, há um mecanismo de checagem de erro, finalizando a construção do pacote. A Figura 24 mostra o formato das mensagens trocadas entre dois dispositivos sob o protocolo de comunicação *Modbus RTU*.

PACOTE GENÉRICO MODBUS RTU					
Início	Endereço	Função	Dados	CRC	Fim
Tempo de espera 3 a 5 chars	8 bits	8 bits	N x 8 bits	16 bits	Tempo de espera 3 a 5 chars

**Figura 24:** Composição do pacote de informações do Modbus RTU

Entre as mensagens é necessário um espaço de tempo de 3 a 5 *chars*, ou seja, o tempo necessário para transmitir de 3 a 5 bytes. O cálculo deve ser realizado baseado na



taxa de transmissão serial escolhida. Os campos que compõem o pacote a se transmitido contém as seguintes informações (FILHO, 2002):

- O campo *endereço* pode assumir valores entre 0 e 247 (em hexadecimal, 0x00 a 0xf7), sendo que os dispositivos são endereçados com valores entre 1 e 247. O endereço zero é reservado para ser utilizado como endereço *broadcast*, ou seja, as mensagens destinadas ao endereço zero são reconhecidas por todos os elementos da rede;
- O campo *função* recebe o código da função que será utilizada para realizar a comunicação. O valor pode variar de 1 a 255 (0x01 a 0xff, em hexadecimal), porém, serão reconhecidos apenas valores que variam entre 1 e 127 (0x01 e 0x7f, em hexadecimal). Isso deve-se ao fato de que o bit mais significativo indica um código de diagnóstico de falha ou sucesso na comunicação;
- O campo *dados* pode variar conforme a função e o tipo da mensagem, requisição ou resposta. No caso de escrita em algum registrador, este campo pode representar o valor que se deseja escrever;
- O campo *CRC (Cyclical Redundancy Check)* contém o resultado de um mecanismo de checagem de erros. O *CRC* verifica o conteúdo de toda a mensagem. Ele é aplicado independentemente de qualquer método de verificação de paridade utilizado para cada um dos caracteres da mensagem.

O *CRC* ocupa dois bytes do pacote, contendo um valor binário de 16 bits. Seu valor é calculado pelo dispositivo de transmissão, que acrescenta o *CRC* à mensagem. O dispositivo receptor refaz a operação durante a recepção da mensagem, e compara o valor calculado para o valor real que recebeu no campo *CRC*. Se os dois valores não forem iguais, ocorrerá um erro e o pacote de dados não será enviado (MODICON, 1996).

O protocolo *Modbus RTU* pode ser utilizado para comunicar o sistema ScadaBR com dispositivos como microcontroladores e CLP's.

