

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO E SISTEMAS
CURSO DE ENGENHARIA DE PRODUÇÃO ELÉTRICA

Erik Spanhol Lind

**Análise comparativa de métodos de roteirização para separação de pedidos em
armazéns**

Florianópolis
2022

Erik Spanhol Lind

**Análise comparativa de métodos de roteirização para separação de pedidos em
armazéns**

Trabalho Conclusão do Curso de Graduação em Engenharia de Produção Elétrica do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito para a obtenção do título em Engenharia, área Elétrica, habilitação Produção Elétrica.
Orientador: Prof. Eduardo Ferreira da Silva

Florianópolis

2022

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Lind, Erik

Análise comparativa de métodos de roteirização para
separação de pedidos em armazéns / Erik Lind ; orientador,
Eduardo Silva, 2022.

83 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Engenharia de Produção Elétrica, Florianópolis,
2022.

Inclui referências.

1. Engenharia de Produção Elétrica. 2. Separação de
pedidos. 3. Picking. I. Silva, Eduardo. II. Universidade
Federal de Santa Catarina. Graduação em Engenharia de
Produção Elétrica. III. Título.

Erik Spanhol Lind

**Análise comparativa de métodos de roteirização para separação de pedidos em
armazéns**

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de Bacharel em Engenharia de Produção Elétrica e aprovado em sua forma final pelo Curso de Engenharia de Produção Elétrica

Florianópolis, 14 de março de 2022.

Prof.(a) Mônica Maria Mendes Luna, Dra.
Coordenador do Curso

Banca Examinadora:

Prof. Eduardo Ferreira da Silva, Dr.
Orientador(a)
Universidade Federal de Santa Catarina

Prof.(a) Mônica Maria Mendes Luna, Dra.
Avaliador(a)
Universidade Federal de Santa Catarina

Prof. Ricardo Vilarroel Dávalos, Dr.
Avaliador(a)
Universidade Federal de Santa Catarina

Este trabalho é dedicado aos meus queridos pais, irmã e irmão.

AGRADECIMENTOS

Gostaria de agradecer à Universidade Federal de Santa Catarina e seu corpo docente por toda a estrutura e oportunidades oferecidas durante a graduação.

Ao meu orientador, Eduardo Ferreira da Silva, por aceitar ser meu orientador, por me introduzir a área de trabalho, pela motivação radiante em torno do assunto e todo auxílio e solicitude ao prestar suporte na construção deste trabalho.

Aos amigos que fiz durante a graduação, com quem convivi intensamente nos últimos anos e que enriqueceram meu caminho com muita troca de experiências, companheirismo e bom humor, em especial ao Alisson Carlos e Lucas Dias,

Agradeço à minha namorada, Laura Olbrich, por todas as vivências, incentivo, companheirismo e dedicação que influenciaram a minha formação.

Aos meus pais, Jerry e Sebastiana, pelo apoio às decisões que tomei, pelo estímulo a enfrentar novos desafios e pelo suporte por eles dado para que tudo isso fosse possível acontecer.

Aos meus irmãos, Hanna e Lukas pelo laço que temos e a vontade de ajudar uns aos outros mesmo distantes fisicamente.

Por fim, agradeço a todos aqueles que de alguma forma estiveram presentes durante a minha caminhada e contribuíram para minha formação. Obrigado!

RESUMO

Os impactos gerados pela pandemia provocaram uma transformação intensa no comportamento de consumo. Houve um enorme incremento na utilização dos meios digitais, devido às novas limitações e restrições impostas. Enquanto muitas empresas diminuíram seu ritmo, as vendas pela internet passaram a aumentar. Em razão disso, a necessidade por centros de distribuições para o e-commerce também aumentou. Dentre as operações do centro de distribuição, a atividade de *picking* é conhecida por ser bastante desafiadora em termos de trabalho, tempo e custo e representa a maior parcela dos custos operacionais envolvidos. Portanto, reduzir o tempo de caminhada/busca na atividade de *picking* pode ser uma maneira eficaz de aumentar a produtividade geral do centro de distribuição. Nesse contexto, o trabalho tem como objetivo propor alternativas de separação de pedidos que minimizem a distância total de viagem no processo de separação de pedidos por meio de otimização e simulação, considerando-se o problema específico do problema geral do caixeiro-viajante, de roteirização para um único operador em um modelo bidimensional do centro de distribuição com layouts e disposição dos itens diferentes. Duas alternativas de movimentação foram escolhidas para alcançar o objetivo. A primeira, conhecida por *wave picking*, simula o impacto na distância total percorrida por diferentes agrupamentos de pedidos. A segunda, clusterização espacial, com o objetivo de agrupar pedidos por clusters geográficos para reduzir a distância máxima entre dois itens em uma mesma lista de pedidos. Constatou-se que a aplicação da técnica de separação de pedidos em ondas em conjunto com a clusterização espacial se mostrou eficaz no sentido de guiar a movimentação de um operador responsável pela atividade de *picking* como alternativa de solução de curto prazo, no qual a solução é apresentada sem tanta exigência de tempo e recursos como por exemplo a realocação dos itens ou reestruturação do layout do centro de distribuição. Após as conclusões, o trabalho apresenta sugestões e recomendações para trabalhos futuros.

Palavras-chave: Otimização. *Picking*. Clusterização Espacial. Centro de Distribuição, Armazém.

ABSTRACT

The impacts generated by the pandemic caused an intense transformation in consumption behavior. There was a huge increase in the use of digital media, due to the new limitations and restrictions imposed. While many companies have slowed down, internet sales have increased. As a result, the need for distribution centers for e-commerce has also increased. Among the distribution center operations, the picking activity is known to be quite challenging in terms of work, time and cost and represents the largest part of the total operational costs involved. Therefore, reducing walking time in picking activity can be an effective way to increase the overall distribution center productivity. In this context, this study aims to propose order picking alternatives that minimize the total travel distance in the order picking process through optimization and simulation, considering the specific problem of the general traveling salesman problem, the single picker routing problem in a two-dimensional model of the distribution center with different layouts and arrangement of items. Two picking alternatives were chosen to achieve the objective. The first, known as wave picking, simulates the impact on the total distance traveled by different order groupings. The second, spatial clustering, aims to generate batches by geographic clusters to reduce the maximum distance between two items in the same order list. It was found that the application of the order picking technique in waves together with spatial clustering proved to be effective in guiding the movement of an operator responsible for the picking activity as an alternative short-term solution, in which the solution is presented without requiring as much time and resources as, for example, the relocation of items or restructuring the layout of the distribution center. After the conclusions, the work presents suggestions and recommendations for future work.

Keywords: Optimization. Picking. Spatial Clustering. Distribution Center, Warehouse.

LISTA DE FIGURAS

Figura 1 - Etapas do estudo.....	29
Figura 2- Exemplo de layout de armazém.....	33
Figura 3 - Bibliotecas para geração de dados.....	34
Figura 4 - Criação das posições de armazenamento e layout do armazém	34
Figura 5 - Geração dos itens do armazém	35
Figura 6 – Bancos de dados com os atributos existentes em um WMS necessários ao trabalho	36
Figura 7 - Exemplo de itens para coleta no armazém	37
Figura 8 - Distância entre dois pontos.....	39
Figura 9 - Exemplo de lista de itens dos pedidos	40
Figura 10 - Exemplo de lista de itens dos pedidos em batches	41
Figura 11 - Distância média entre dois pontos	43
Figura 12 - Função para determinação de clusters	45
Figura 13 - Função para determinação da distância caminhada.....	46
Figura 14 - Modelo simplificado da segunda estratégia.....	48
Figura 15 - Bibliotecas para geração de dados.....	49
Figura 16 - Criação das posições de armazenamento e layout do armazém	50
Figura 17 - Geração dos itens do armazém	50
Figura 18 - Resumo do dataframe do Layout 1.....	51
Figura 19 - Esboço do Layout 1	52
Figura 20 - Definição das restrições da lista de pedidos	53
Figura 21 - Geração da lista de pedidos	54
Figura 22 - Resumo da lista de pedidos da análise 1.....	55
Figura 23 - Resultado em colunas da primeira análise do Layout 1	56
Figura 24 - Resultado da primeira análise do Layout 1 com demanda alterada	58
Figura 25 - Resultado em colunas da segunda análise do Layout 1	59
Figura 26 - Ajuste de limiar de distância da segunda análise do Layout 1	60
Figura 27 - Ajuste de limiar de distância da segunda análise do Layout 1 com demanda alterada.....	61
Figura 28 - Resultados das estratégias no Layout 1	62
Figura 29 - Resultados das estratégias no Layout 1 com demanda alterada	63

Figura 30 - Ajuste de onda com melhor limiar de distância do cenário 1.....	64
Figura 31- Esboço do Layout 2	65
Figura 32 - Resultado em colunas da primeira análise do Layout 2	66
Figura 33 - Resultado da primeira análise do Layout 2 com demanda alterada	68
Figura 34 - Resultado em colunas da segunda análise do Layout 2.....	69
Figura 35 - Ajuste de limiar de distância da segunda análise do Layout 2	70
Figura 36 - Ajuste de limiar de distância da segunda análise do Layout 2 com demanda alterada.....	71
Figura 37 - Ajuste de onda com melhor limiar de distância do cenário 2.....	72
Figura 38 - Resultados das estratégias no Layout 2	73
Figura 39 - Resultados das estratégias no Layout 1 com demanda alterada	74
Figura 40 - Resultado global da aplicação das estratégias	76

LISTA DE TABELAS

Tabela 1 - Técnicas para análise de picking no armazém	38
Tabela 2 - Formato de resposta da análise 1	42
Tabela 3 - Resultado numérico da primeira análise do Layout 1	56
Tabela 4 - Resultado detalhado da primeira análise do Layout 1	57
Tabela 5 - Resultado numérico da segunda análise do Layout 1	59
Tabela 6 - Dados do ajuste de limiar de distância do Layout 1.....	60
Tabela 7 - Resultado detalhado da primeira análise do Layout 2	67
Tabela 8 - Dados do ajuste de limiar de distância do Layout 2.....	70
Tabela 9 - Resultados da estratégia de wave picking.....	75
Tabela 10 - Resultados da estratégia de clusterização espacial.....	75

LISTA DE ABREVIATURAS E SIGLAS

CD – Centro de Distribuição

TSP – *Traveling Salesman Problem*

SPRP – *Single Picker Routing Problem*

WMS – *Warehouse Management System*

OMS – *Order Management System*

TMS – *Transport Management System*

CSV – *Comma-separated Values*

SKU – *Stock Keeping Unit*

SUMÁRIO

1.	INTRODUÇÃO	17
1.1.	JUSTIFICATIVA.....	18
1.2.	OBJETIVOS	19
1.2.1.	OBJETIVO GERAL	19
1.2.2.	OBJETIVOS ESPECÍFICOS.....	19
1.3.	DELIMITAÇÕES	19
2.	FUNDAMENTAÇÃO TEÓRICA	20
2.1.	ARMAZÉM	21
2.1.1.	AGRUPAMENTO DE LOCALIZAÇÕES DE <i>PICKING</i> POR <i>CLUSTERS</i>	22
2.1.2.	ATIVIDADE DE <i>PICKING</i>	22
2.1.3.	<i>BATCHING</i>	23
2.1.4.	<i>PICKING</i> POR ONDAS	24
2.1.5.	POLÍTICAS DE ARMAZENAGEM	24
2.1.6.	SEPARAÇÃO DE PEDIDOS.....	25
2.1.7.	<i>SINGLE PICKER ROUTING PROBLEM</i> (SPRP).....	26
2.1.8.	SISTEMAS DE SEPARAÇÃO DE PEDIDOS	26
2.1.9.	SISTEMA WMS	27
2.1.10.	TRAVELING SALESMAN PROBLEM (TSP).....	28
3.	MÉTODOS ADOTADOS	29

3.1.	GERAÇÃO DE DADOS	30
3.2.	MODELAGEM.....	30
3.3.	AVALIAÇÃO DE CENÁRIOS.....	31
4.	DESENVOLVIMENTO	32
4.1.	GERAÇÃO DE DADOS DO ARMAZÉM.....	32
4.1.1.	GERAÇÃO DO LAYOUT DE UM ARMAZÉM.....	32
4.1.2	GERAÇÃO DA LISTA DE PEDIDOS	33
4.1.3	GERAÇÃO DE DADOS DE UM SISTEMA DE GERENCIAMENTO ARMAZÉNS	35
4.2	IDENTIFICAÇÃO DE ESTRATÉGIAS DO SPRP	37
4.3	ESTRATÉGIA DE <i>WAVE PICKING</i>	38
4.3.1	FUNÇÕES PARA APLICAÇÃO DA ANÁLISE DE <i>WAVE PICKING</i>	39
	4.3.1.1 Função para cálculo da mínima distância entre dois pontos	39
	4.3.1.2 Função para busca do próximo item a ser coletado da lista de pedidos 40	
	4.3.1.3 Função para criação das ondas de coleta.....	41
	4.3.1.4 Função para listagem da rota criada para coleta de onda.....	41
4.3.2	SIMULAÇÃO DA ANÁLISE DE <i>WAVE PICKING</i>	42
4.4	ESTRATÉGIA DE <i>CLUSTERING</i> ESPACIAL.....	42
4.4.1	CONSIDERAÇÕES DOS PEDIDOS SINGULARES E PEDIDOS MÚLTIPLOS 43	
4.4.2	CONSIDERAÇÕES PARA CÁLCULO DAS DISTÂNCIAS	44
4.4.3	<i>CLUSTERIZAÇÃO</i>	44

4	SIMULAÇÃO DA ANÁLISE DE <i>CLUSTERIZAÇÃO</i> ESPACIAL.....	48
5	RESULTADOS.....	49
5.1	CENÁRIOS PARA SIMULAÇÃO	49
5.1.1	GERAÇÃO DOS ITENS E SUAS LOCALIZAÇÕES NO CENÁRIO 1.....	49
5.1.2	GERAÇÃO DA LISTA DE PEDIDOS DO CENÁRIO 1	52
5.1.3	PRIMEIRA ANÁLISE DO CENÁRIO 1	55
	5.1.3.1 Primeira análise do cenário 1 com demanda alterada	57
5.1.4	SEGUNDA ANÁLISE DO CENÁRIO 1	58
	5.1.4.1 Segunda análise do cenário 1 com demanda alterada	61
5.1.5	COMPARAÇÃO DAS ANÁLISES DO CENÁRIO 1.....	62
5.2.1	GERAÇÃO DOS ITENS E SUAS LOCALIZAÇÕES NO CENÁRIO 2.....	64
5.2.2	GERAÇÃO DA LISTA DE PEDIDOS DO CENÁRIO 2	65
5.2.3	PRIMEIRA ANÁLISE DO CENÁRIO 2.....	65
	5.2.3.1 Primeira análise do cenário 2 com demanda alterada	67
5.2.4	SEGUNDA ANÁLISE DO CENÁRIO 2.....	68
	5.2.4.1 Segunda análise do cenário 2 com demanda alterada	71
5.2.5	COMPARAÇÃO DAS ANÁLISES DO CENÁRIO 2.....	72
5.3	AVALIAÇÃO.....	74
5.3.1	AVALIAÇÃO DA ESTRATÉGIA DE <i>WAVE PICKING</i>	74
5.3.2	AVALIAÇÃO DA ESTRATÉGIA DE <i>CLUSTERIZAÇÃO</i> ESPACIAL	75
5.3.3	AVALIAÇÃO GLOBAL.....	75
6	CONCLUSÃO	78

REFERÊNCIAS.....80

APÊNDICE A – REPOSITÓRIO COM OS CÓDIGOS DAS ANÁLISES DO LAYOUT 1 E

2 83

1. INTRODUÇÃO

No Brasil, o e-commerce atingiu um recorde histórico de vendas no primeiro semestre de 2021 com um total superior a 53 bilhões de reais, representando um crescimento de 31% em relação ao mesmo período do ano anterior (EBIT,2021). Ou seja, nos últimos tempos, o modelo de compras tradicional, onde os clientes fazem a maior parte do trabalho intensivo de mão de obra (como coleta de pedidos e transporte das mercadorias para casa), está perdendo espaço para o atendimento eletrônico, onde os varejistas devem entregar pedidos personalizados para locais altamente dispersos dentro de janelas de tempo relativamente estreitas (EDWARDS; MCKINNON; CULLINANE, 2010).

Isso significa que a necessidade do uso de armazéns também aumentou pois estes contribuem para a eficiência do e-commerce e, para acompanhar a crescente concorrência na indústria e superar os concorrentes, os armazéns devem realizar todas as suas operações da maneira mais eficiente para garantir o funcionamento da cadeia de suprimentos e minimizar o custo (SHETTY; SAH; CHUNG, 2020).

Dentre as operações do armazém, o *picking* é conhecido por ser bastante desafiador em termos de trabalho, tempo e custo, e pode representar de 50 a 75% do custo operacional total para um armazém típico (SHETTY; SAH; CHUNG, 2020). Sendo assim, reduzir o tempo de caminhada/busca na atividade de *picking* pode ser uma maneira eficaz de aumentar a produtividade geral do centro de distribuição.

Nesse contexto, o presente trabalho tem como objetivo avaliar as alternativas de movimentação do operador dentro do armazém considerando uma alternativa específica do problema geral do caixeiro-viajante (“Traveling Salesman Problem-TSP”), o problema de roteirização para um único operador (“Single Picker Routing Problem-SPRP”) em um modelo bidimensional do armazém, onde a partir de uma lista de locais de armazenamento e suas respectivas coordenadas, a análise leva em conta diferentes técnicas para a determinação da separação de pedidos de forma a encontrar soluções eficientes para guiar a movimentação do operador de acordo com o layout do armazém e o tamanho da lista de pedidos e, conseqüentemente, minimizar o tempo e a distância total percorrida pelo operador ao coletar os itens contidos na lista de pedidos para expedição no armazém.

1.1. JUSTIFICATIVA

A globalização em expansão e o avanço da economia mundial levaram a um aumento da demanda por serviços de logística, transporte e armazenamento. E, o ambiente de negócios se tornou altamente competitivo, os armazéns não servem apenas como um local para armazenar estoque, mas também para gerenciar e operar serviços de valor agregado, os armazéns estão constantemente sob pressão para aumentar a produtividade e a precisão, reduzindo custos e melhorando o atendimento ao cliente (KARIM et al., 2021).

Em paralelo à necessidade de aumentar a produtividade dos armazéns, a tecnologia da informação passou por grandes transformações, e sua aplicação serve como ferramenta para apoiar a obtenção, processamento e distribuição de dados, o que torna mais eficiente o processo de tomada de decisões por parte da gestão de um armazém. Atualmente, um fator muito competitivo é o tempo de fluxo de informações, que encurta os processos nas empresas (KOLINSKI; SLIWZYNSKI, 2015).

A separação de pedidos tem sido identificada há muito tempo como a atividade mais trabalhosa e cara para quase todos os armazéns; o custo de separação de pedidos é estimado em até 55% da despesa operacional total do armazém. Qualquer baixo desempenho na separação de pedidos pode levar a um serviço insatisfatório e alto custo operacional para o armazém e, conseqüentemente, para toda a cadeia de suprimentos. Para operar com eficiência, o processo de separação de pedidos precisa ser projetado de forma robusta e controlado de forma otimizada (DE KOSTER; LE DUC; ROODBERGEN, 2007).

Diante disto, a análise comparativa de métodos de roteirização para separação de pedidos em armazéns, tema de estudo do presente trabalho, traz atenção ao abordado acima, conectando a necessidade de aumentar a produtividade de armazéns com o uso de tecnologias, com o intuito de avaliar formas de trazer mais eficiência na atividade de picking, utilizando-se modelos matemáticos processados computacionalmente, o que conseqüentemente pode aumentar a produtividade geral do armazém ao auxiliar operadores responsáveis pela atividade de picking a partir de um método eficiente para separação de pedidos com o objetivo de reduzir a distância total percorrida nas coletas.

1.2. OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste TCC.

1.2.1. Objetivo Geral

Avaliar alternativas de movimentação de operadores no processo de *picking* em diferentes layouts de centros de distribuição.

1.2.2. Objetivos Específicos

- Criar um banco de dados somente com os atributos existentes em um WMS que sejam necessário ao trabalho;
- Identificar e selecionar algumas estratégias do problema de roteirização para um único operador (“Single Picker Routing Problem-SPRP”);
- Projetar um modelo de simulação para estimar o impacto dessas técnicas em centros de distribuição com layouts diferentes;
- Produzir uma avaliação global em função da demanda esperada, das técnicas selecionadas e dos diferentes layouts dos centros de distribuição.

1.3. DELIMITAÇÕES

Existem diversas formas e métodos utilizados para otimizar as atividades de um armazém, a alteração da disposição dos itens, por exemplo, afeta diretamente o processo de recebimento e armazenamento de mercadorias, bem como o processo de separação de pedidos. Porém, exige bastante planejamento, uso de recursos e tempo até sua implementação e, a necessidade por mudanças pode ocorrer de maneira muito rápida. Isso faz com que, muitas vezes, a alternativa de alterar a disposição dos itens seja inviável, pois, no momento em que as mudanças ocorreram, a demanda pode ter sido alterada novamente e essa medida não atende mais de maneira eficiente a demanda do momento atual. Partindo disso, o presente trabalho se

propõe a realizar uma análise comparativa de métodos de roteirização para separação de pedidos em armazéns, no qual um operador, responsável pela atividade de *picking* possa utilizar soluções mais rápidas de modo a retornar cálculos eficazes das distâncias entre as posições das mercadorias dentro de um armazém a fim de criar estratégias de movimentação na coleta de mercadorias que servirá como guia para a separação dos pedidos. As técnicas analisadas não consideram operadores atuando em paralelo, somente um operador é considerado e não se almeja realizar análise dos custos envolvidos.

1.4. ESTRUTURA DO TRABALHO

O trabalho está estruturado em seis capítulos, abordando as ideias de maneira sequencial para facilitar a compreensão. O primeiro capítulo é iniciado com a introdução ao tema, contextualização do assunto bem como os objetivos e delimitações do trabalho. O segundo capítulo é composto pela fundamentação teórica, trazendo conceitos e explicações necessárias para compreensão do trabalho. No capítulo seguinte é feita a abordagem dos métodos adotados no estudo, onde a partir dos direcionamentos estabelecidos é possível obter os desenvolvimentos obtidos no capítulo subsequente. Neste capítulo os tópicos levantados no capítulo anterior são explorados. O quinto capítulo demonstra os resultados de todas as análises dos cenários escolhidos para simulação, obtidos a partir da implementação das estratégias estabelecidas no capítulo anterior. Por fim, o sexto e último capítulo traz as conclusões obtidas com a realização do estudo, bem como recomendações, sugestões e oportunidades para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo traz uma revisão da literatura relacionada aos conceitos centrais do assunto explorado para dar continuidade ao entendimento do trabalho. Inicialmente, são apresentados conteúdos relacionados à definição mais ampla de um armazém até aprofundar-se no processo de *picking*, problemas para determinação da menor rota e as variáveis necessárias que possibilitam o alcance do objetivo proposto, a partir do design do modelo de simulação.

2.1. ARMAZÉM

Nos últimos anos os armazéns passaram por transformações, tornando-se mais estratégicos, complexos e, cada vez mais, impulsionados a adotarem a digitalização de processos. Os armazéns mais modernos são projetados com base em diversos estudos de otimização. Em consequência disso, os Sistemas de Gerenciamento de Armazém (WMS) tornam-se extremamente importantes e mais complexos(KARASEK, 2013).

Para Karasek (2013) os processos básicos dos armazéns são o recebimento, armazenamento, arrumação, separação e envio de mercadorias, com possível desdobramento em subtarefas, principalmente quando aplicam-se agrupamentos ou zoneamentos, verificação de pedidos quanto à sua integridade, embalagem e expedição. De maneira mais simplificada, Zunic et al (2018) desdobra a estrutura operacional e organizacional e os processos de coordenação dos armazéns em quatro etapas: recebimento, armazenamento, recuperação e expedição de mercadorias.

Como dito por Karasek (2013), os sistemas de gerenciamento de armazém (WMS) se tornaram cada vez mais importantes devido a modernização dos armazéns. O que vai ao encontro do que dizem Zunic et al. (2018) que reforçam que um sistema de gerenciamento de armazém (WMS) é constituído por diversos desafios que precisam ser resolvidos para melhorar o processo de trabalho do armazém. Um WMS de alta qualidade pode melhorar o processo de negócios e reduzir custos. E, é extremamente importante para os trabalhadores pois, mesmo em armazéns menores, os operadores não são capazes de tomar decisões ideais em todas as situações.

Em síntese, pode-se dizer que os armazéns são estruturas físicas que armazenam mercadorias com a finalidade de despachá-las e servem como uma ligação entre fornecedor e consumidor com a função de garantir maior eficiência e agilidade na entrega de mercadorias em busca da maior satisfação do cliente.

2.1.1. Agrupamento de localizações de *Picking* por *Clusters*

Há um número cada vez maior de métodos disponíveis para a análise de agrupamento espacial. Essas técnicas podem ser divididas em duas categorias: aquelas que são utilizadas para determinar se o *clustering* está presente na região de estudo e aquelas que procuram identificar a localização dos *clusters*. A primeira categoria de testes é chamada de técnicas de agrupamento global e esses métodos fornecem uma única estatística que resume o padrão espacial da região. O segundo tipo de metodologia é chamado de *clustering* local. Os métodos locais examinam sub-regiões dentro do estudo para determinar se aquela área representa um grupo de valores altos (*hot spot*) ou valores baixos (*cold spot*) ([ALDSTADT, 2010](#)).

O *clustering* espacial é a tarefa de agrupar um conjunto de pontos de forma que os objetos no mesmo *cluster* sejam mais semelhantes entre si do que os objetos em outros *clusters*, sendo assim, o agrupamento dos locais de *picking* pode ser aplicado para reduzir a distância percorrida pelo operador nas rotas definidas para o *picking* ([SACI, 2021](#)).

2.1.2. Atividade de *Picking*

O *picking* é um processo que cobre muitos problemas. Inicialmente, uma lista de pedidos é fornecida a um operador. A coleta leva cerca de 50% a 70% dos custos operacionais do CD e consiste em: Viagem (55%), Pesquisa (15%), Extração (10%), e Papelada (20%) (KOSTER et al, 2007). O *picking* pode ser de dois tipos, homogêneo e heterogêneo. O *picking* homogêneo é bastante simples, o selecionador opera simplesmente com um palete inteiro. Na separação heterogênea, o selecionador é informado onde e o que escolher, em que quantidade e unidades. Devido às atuais necessidades do cliente, o *picking* heterogêneo é logicamente o mais frequente (KARASEK, 2013).

O recente crescimento dramático do e-commerce dá origem ao aumento exponencial das operações de *picking* e, portanto, tem recebido uma atenção considerável tanto da academia quanto da indústria (KOSTER et al., 2007). É importante observar que entre todas as atividades relacionadas à coleta de pedidos, a viagem do operador para realizar a coleta dos itens solicitados nos pedidos ocupa a maior parcela do tempo gasto, ou seja, a minimização no tempo

gasto com movimentação do operador contribui para o aumento da eficiência do CD em geral (SHETTY et al., 2020).

De acordo com Saci (2020), o *Single Picker Routing Problem* (SPRP) é usado para determinar a rota mínima no processo de separação para preparar um ou vários pedidos, é uma aplicação específica do problema geral do *Traveling Salesman Problem* (TSP), ou Problema do Caixeiro-Viajante, respondendo à pergunta: "Dada uma lista de locais de armazenamento e as distâncias entre cada par de locais, qual é a rota mais curta possível que visita cada local de armazenamento e retorna ao depósito?".

Em concordância com o afirmado, Koster et al. (2007) abordam que este tipo de problema, de minimizar o tempo de viagem de coleta de pedidos, é usado para determinar a rota mínima no processo de separação para preparar um ou vários pedidos e é geralmente formulado como um caso especial de problema do caixeiro viajante.

2.1.3. *Batching*

Quando os pedidos são volumosos, cada pedido pode ser separado individualmente, ou seja, a cada saída do operador para coleta. Por outro lado, segundo De Koster et al (2007) quando os pedidos são pequenos, existe um potencial para reduzir os tempos de viagem, escolhendo um conjunto de pedidos em um único roteiro de picking. O *batching* é o método de agrupar uma lista de pedidos em um número de subconjuntos de pedidos, no qual cada pedido do subconjunto pode ser coletado em uma mesma saída de *picking*, que é a saída para coletar as mercadorias no armazém.

De acordo com Henn et al. (2012), a consolidação de pedidos pode ser organizada de duas formas distintas, uma considerada a separação discreta dos pedidos, no qual cada saída na atividade de coleta dos itens compreende os itens apenas de um único cliente, enquanto na separação por *batches*, os itens de vários clientes podem ser coletados simultaneamente em uma única saída para coleta.

2.1.4. *Picking* por Ondas

O *picking* por onda é aplicado quando operadores que coletam os itens dos pedidos lidam com um conjunto maior de pedidos e o tempo de resposta para o conjunto completo é importante, significa que uma série de lotes, uma onda, é separada para coleta em uma mesma saída de *picking* (GADEMANN et al, 2001).

Assumindo uma estrutura comumente aplicada em armazéns voltados ao e-commerce, nos quais os itens são armazenados em prateleiras de 4 níveis, organizadas em várias fileiras e corredores, Saci (2020) considera que a primeira maneira intuitiva de otimizar o processo de *picking* neste armazém seria combinar pedidos em uma mesma rota de coleta de maneira a reduzir a distância total percorrida, ao invés do *picking* realizado exclusivamente para cada pedido - esta estratégia é comumente chamada de *picking* por onda.

2.1.5. Políticas de armazenagem

Os produtos precisam ser colocados em locais de armazenamento antes que eles possam ser separados para atender aos pedidos dos clientes. É necessário determinar qual será a política de armazenamento dentro do armazém onde um conjunto de regras determinará como os produtos devem ser armazenados (DE KOSTER; LE DUC; ROODBERGEN, 2007).

De acordo com De Koster et al. (2007) existem diversas formas de realizar a atribuição dos locais de armazenamento no armazém. Sendo as mais comuns: armazenamento aleatório, armazenamento mais próximo em aberto, armazenamento dedicado, armazenamento de rotatividade total e o armazenamento baseado em classe. No armazenamento aleatório cada palete de entrada recebe um local no armazém que é selecionado aleatoriamente entre todos os locais vazios elegíveis para o armazenamento com a mesma probabilidade. Já no armazenamento mais próximo em aberto, o produto será armazenado no primeiro local vazio encontrado pelo operador. No armazenamento dedicado, cada produto será destinado a um local fixo no armazém. Algumas das vantagens são que o operador se torna familiarizado com a localização dos produtos e, no caso de os produtos apresentarem diferentes pesos é possível alocar seus locais de armazenamento de maneira em que a sequência escolhida facilite o

trabalho do operador e reduza seus esforços. Por outro lado, uma das desvantagens é que um espaço será reservado a um produto mesmo quando este estiver em falta de estoque. O quarto tipo mostrado, o armazenamento por rotatividade total, conhecido por *full turn-over storage*, essa política de armazenamento distribui os produtos no armazém baseado nas saídas, sendo assim, produtos com taxa de vendas mais altas estarão armazenados em locais mais acessíveis, geralmente próximo ao depósito, enquanto os produtos com pouca saída, geralmente, estarão armazenados na região traseira do armazém. Por fim, no armazenamento baseado em classe, cada classe é atribuída a uma área dedicada do armazém, no qual o armazenamento dentro uma área é aleatória e as classes são determinadas por medidas relacionadas a frequência de demanda dos produtos (DE KOSTER; LE DUC; ROODBERGEN, 2007).

2.1.6. Separação de pedidos

A separação de pedidos envolve o processo de agrupamento e programação dos pedidos dos clientes, atribuição de estoque a seus locais para a lista de pedidos, liberação de pedidos, separação dos artigos dos locais de armazenamento e disposição dos artigos separados. Os pedidos do cliente consistem em linhas de pedido, cada linha para um produto exclusivo ou unidade de manutenção de estoque (SKU), em uma determinada quantidade. É possível encontrar diversos tipos de sistemas de separação de pedidos nos armazéns, muitas vezes múltiplos sistemas de separação de pedidos podem ser aplicados dentro de um mesmo armazém, por exemplo, em diferentes zonas de um armazém os sistemas de separação de pedidos pode variar de acordo com o uso de humanos ou máquinas automatizadas utilizadas. Entretanto, a maioria dos armazéns emprega humanos para a separação de pedidos, dentre estes, os sistemas *picker-to-parts*, onde o selecionador de pedidos caminha ou dirige pelos corredores para separar os itens, são os mais comuns (DE KOSTER; LE DUC; ROODBERGEN, 2007).

De Koster (2007) sugere também que um dos objetivos principais dos sistemas de separação de pedidos é maximizar o nível de serviço sujeito a restrições de recursos como mão de obra, máquinas e capital. O nível de serviço é composto por uma variedade de fatores, como média e variação do tempo de entrega, integridade e precisão do pedido. Um elo crucial entre a separação de pedidos e o nível de serviço é que quanto mais rápido um pedido pode ser coletado,

mais cedo ele estará disponível para envio ao cliente. Se um pedido perder o prazo de envio, pode ter que esperar até o próximo período de envio. Além disso, tempos curtos de coleta de pedidos implicam em alta flexibilidade no tratamento de alterações tardias nos pedidos. Minimizar o tempo de separação de pedidos é, portanto, uma necessidade de qualquer sistema de separação de pedidos.

2.1.7. *Single Picker Routing Problem (SPRP)*

De acordo com Saci (2020), o SPRP é uma aplicação específica do problema geral do caixeiro viajante (TSP), usado para determinar a rota mínima na atividade de *picking* para preparar um ou vários pedidos. Conforme dito, Henn (2016) também se refere ao problema de roteirização do SPRP como um caso especial do TSP, no qual a sequência em que os itens devem ser coletados são determinadas e, pode ser definido da seguinte forma: Dado um conjunto de itens a serem coletados com locais de armazenamento conhecidos, em qual sequência os locais devem ser visitados de modo que o comprimento total da rota percorrida pelo operador seja minimizado?

Choueiri (2018) sintetiza dizendo que o *SPRP* procura encontrar a rota de menor distância total, dado um CD e uma lista de posições com localizações de itens a serem coletados, de maneira que partindo de um ponto inicial, todos os locais com itens sejam visitados uma única vez.

2.1.8. Sistemas de separação de pedidos

De acordo com Peerlinck et al. ([s.d]) existem vários sistemas de separação de pedidos. *Picker-to-parts* e *parts-to-picker* são os dois grupos principais. A escolha entre um desses sistemas é bastante complexa e depende de diversos fatores, onde as principais questões que afetam o projeto de um sistema de separação de pedidos normalmente são: propriedades do

material, restrições econômicas, restrições ambientais, requisitos do sistema, estratégias operacionais entre outros.

Similar ao explicado por Peerlinck et al. ([s.d]), De Kooster (2012) diz que múltiplos métodos para sistema de separação de pedidos são adotados, mas a grande maioria emprega humanos para a separação dos pedidos. Dentre eles, o sistema *picker-to-parts* é o mais comum, onde o operador caminha até os itens. Os sistemas *parts-to-picker* incluem sistemas automatizados de armazenamento e recuperação, principalmente usando guindastes de corredor que recuperam uma ou mais unidades de carga (caixas: sistema *miniload* ou paletes) e a trazem para uma posição de coleta. Nesta posição, o selecionador pega o número de peças exigidas pelo pedido do cliente, em seguida a carga restante é armazenada novamente, outros sistemas usam módulos de elevação vertical outros sistemas usam módulos de elevação vertical que também oferecem cargas unitárias ao operador, que é responsável por retirar a quantidade certa de itens. E, o formato mais comum no sistema *picker-to-parts* incluem as variantes básicas de picking por artigo (às vezes chamado de picking por *batches*) ou picking por pedido. No caso de picking por *batches*(lotes), vários pedidos de clientes são separados simultaneamente por um selecionador.

2.1.9. Sistema WMS

A instalação de tecnologias nos CDs tais como o Warehouse Management System (WMS), ou Sistema de Gerenciamento de Armazéns (SGA), se tornou um modo de garantir uma vantagem competitiva em armazenagem ([MACHADO; SELBITTO, 2011](#)).

Na logística de armazenagem e distribuição de mercadorias, o uso da tecnologia da informação possibilitou a redução no tempo de resposta e um aumento na eficiência no uso do espaço físico e na movimentação de materiais e consequentemente, aumento de competitividade (GOMES e RIBEIRO, 2004; NOVAES, 2004).

As principais funções de um WMS são recebimento de pedidos, separação de itens, acompanhamento de embarques, gestão de estoques e geração de informação para decisão gerencial, o que é facilitado pela integração com sistemas de gerenciamento de pedidos (OMS) e sistemas de gerenciamento de transporte (TMS) (KIM et al., 2008).

Dessa forma, o WMS é essencial para a coleta de informações necessárias sobre os pedidos e mercadorias armazenadas nos CDs, para possibilitar agilizar o fluxo das informações, auxiliando a operacionalidade e otimizando o processo de armazenagem (POON et al., 2009). O que permite também identificar endereços no armazém que minimizem as rotas de depósito e de coleta, de acordo com a variedade da demanda dos consumidores, otimizando o uso dos recursos de movimentação e minimizando o tempo até o embarque da mercadoria até o cliente (LING-FENG e LIHUI, 2006; SHIAU, J. e LEE, 2010).

2.1.10. Traveling Salesman Problem (TSP)

O objetivo das políticas de roteamento é sequenciar os itens na lista de separação para garantir um bom percurso pelo armazém. O problema de rotear os selecionadores de pedidos no armazém é, na verdade, um caso especial do Problema do Caixeiro Viajante. Esse problema do caixeiro viajante deve seu nome ao problema descrito pela seguinte situação: um vendedor, começando em sua cidade natal, precisa visitar várias cidades exatamente uma vez e voltar para casa. Ele sabe a distância entre cada par de cidades e quer determinar a ordem em que deve visitar as cidades de modo que a distância total percorrida seja a menor possível. Claramente, a situação do caixeiro viajante tem muitas semelhanças com a de um selecionador de pedidos em um armazém. O selecionador de pedidos começa no depósito (cidade de origem), onde recebe uma lista de coleta, deve visitar todos os locais de coleta (cidades) e, finalmente, retornar ao depósito (DE KOSTER; LE DUC; ROODBERGEN, 2007).

Grande parte da atenção que o problema do TSP tem recebido é porque é um problema relativamente simples de descrever e ainda assim um difícil (do ponto de vista da complexidade) problema de otimização para resolver. No entanto, existem casos importantes de problemas práticos que podem ser formulados como problemas TSP e muitos outros problemas são generalizações deste problema. O TSP pode ser encontrado em diversas áreas como na perfuração de placas de circuitos impressos, na análise da estrutura dos cristais, na revisão de motores de turbina a gás, no manuseio de materiais em um armazém, no agrupamento de matrizes de dados, no sequenciamento de tarefas em uma única máquina, entre outros (HOFFMAN; PADBERG, 2001).

3. MÉTODOS ADOTADOS

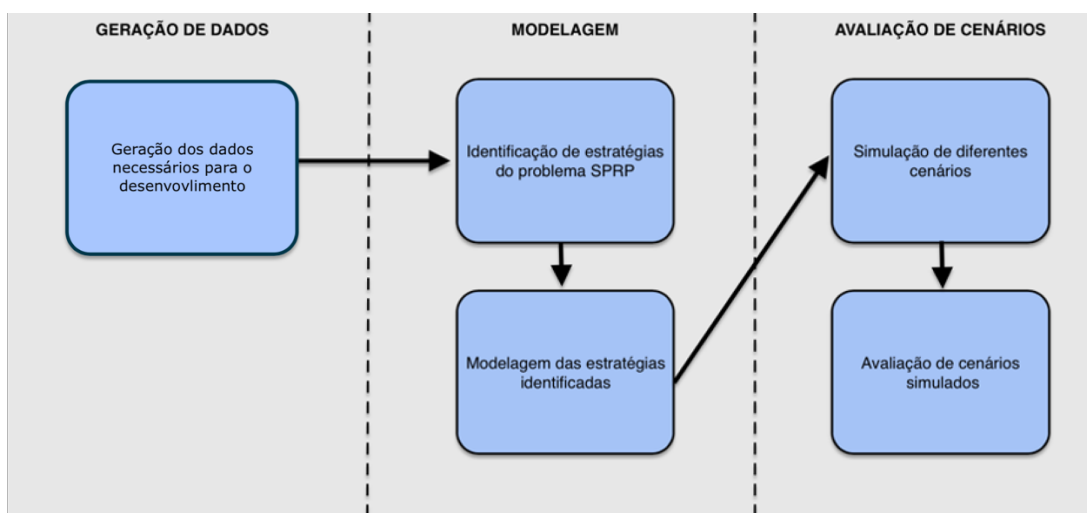
O trabalho está relacionado a resolução de problemas reais envolvendo situações de tomada de decisão, a partir de modelos matemáticos processados computacionalmente.

Busca-se introduzir racionalidade nos processos de tomada de decisão, sem descuidar dos elementos subjetivos e de enquadramento organizacional que caracterizam os problemas, e, portanto, se enquadra na área de pesquisa operacional, sendo modelagem, simulação e otimização sua subárea (ABEPRO, [s.d.]). A abordagem do trabalho é quantitativa e sua análise é baseada em dados primários.

O trabalho teve como base as simulações desenvolvidas por Saci (2020), no qual dados diferentes foram gerados, tanto para geração da lista de pedidos como para o layout do armazém e as adaptações necessárias foram realizadas para a realização da análise comparativa do presente trabalho.

Para o desenvolvimento deste trabalho três macro etapas estruturam o projeto. As etapas estão ilustradas na Figura 1 para melhor compreensão das fases envolvidas no trabalho.

Figura 1 - Etapas do estudo.



Fonte: Figura do autor (2021)

Em seguida são apresentadas as descrições de cada uma das etapas.

3.1. GERAÇÃO DE DADOS

O presente trabalho é realizado baseado na análise comparativa de métodos de roteirização para separação de pedidos em armazéns a partir da linguagem de programação Python. A primeira etapa do trabalho consiste em gerar os dados comumente disponíveis no *WMS* de um armazém. Sendo assim, os dados gerados são caracterizados pelos seguintes tópicos:

- Definição de sistema de separação de pedidos, política de armazenagem e layout do armazém;
- Locais de armazenamento de mercadorias mapeados em $2D(x,y)$ com base no layout do armazém;
- Itens disponíveis no armazém e sua respectiva localização no layout simulado;
- Geração da lista de pedidos dos itens contidos no armazém simulado.

3.2. MODELAGEM

Para iniciar a modelagem é necessário elencar estratégias que possam reduzir a distância total percorrida por um operador no armazém ao realizar a atividade de *picking*. De acordo com Saci (2020), duas estratégias com grande potencial de reduzir significativamente a distância total percorrida pelo operador na atividade de *picking* são as estratégias de *picking* por ondas e *picking* por clusterização espacial.

Em seguida, a modelagem é feita a partir da extração dos dados do banco de dados do *WMS* simulado, coletando a lista de pedidos e os locais de armazenamento. Esses dados são unidos para vincular cada linha de pedido da lista a um local de armazenamento e sua coordenada (x, y) no armazém, o que permite associar cada pedido a uma posição geográfica no espaço físico simulado e serve de base para a simulação do trabalho. Após essa etapa, cada estratégia selecionada será simulada com base em funções necessárias para sua implementação.

3.3. AVALIAÇÃO DE CENÁRIOS

Diferentes layouts do armazém serão considerados bem como o tamanho da lista de pedidos para simulação dos impactos na distância total percorrida. Cada estratégia envolve variáveis que serão “modificadas” ao longo da simulação a fim de avaliar o impacto no output que representa a distância total percorrida pelo operador na atividade de *picking*.

4. DESENVOLVIMENTO

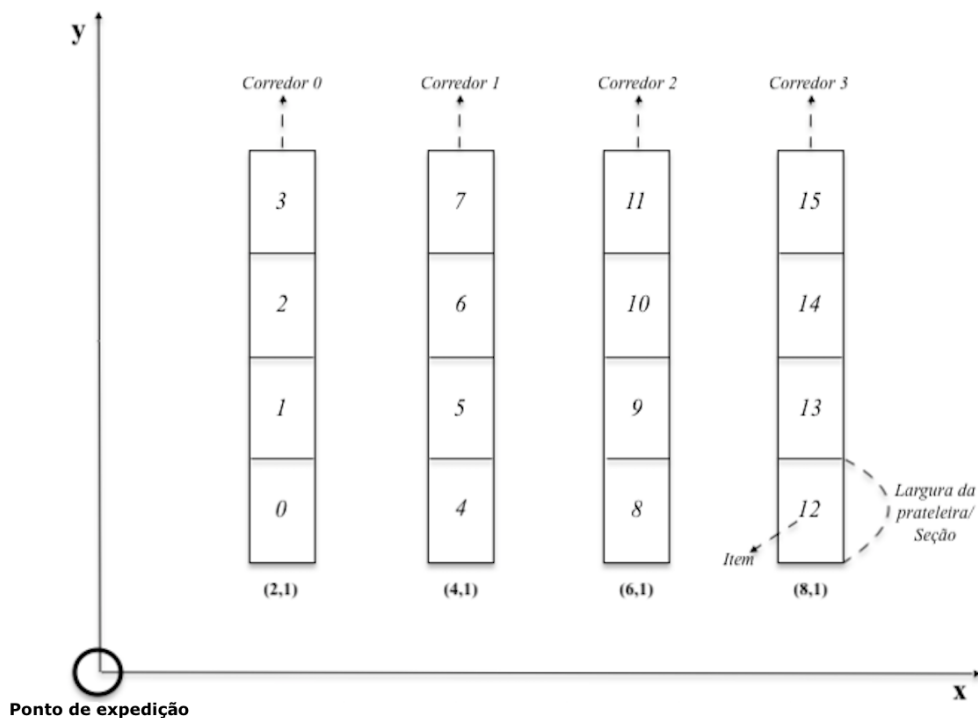
4.1. GERAÇÃO DE DADOS DO ARMAZÉM

O trabalho se inicia com a geração de dados que representam o layout do armazém, a quantidade, posição e uma lista de pedidos. É considerado também que a política de armazenamento do armazém é a de armazenamento dedicado, ou seja, a mercadoria ocupa um local fixo dentro do armazém e, o sistema de separação de pedidos é *picker-to-parts*, no qual um operador caminha até o local onde o item contido na lista de pedidos está armazenado.

4.1.1. GERAÇÃO DO LAYOUT DE UM ARMAZÉM

As principais características determinadas para a definição da organização física do armazém foram as quantidades e distâncias entre corredores/fileiras, seções por corredor, prateleiras por seções, quantidade de itens distintos em uma mesma posição, quantidade total de itens no armazém e posição do ponto de expedição.

Figura 2- Exemplo de layout de armazém



Fonte: Adaptado de Saci (2020)

A Figura 2 ilustra um exemplo de layout de armazém com as seguintes características:

- Distância entre corredores: 2m;
- Largura da prateleira: 1m;
- Quantidade de corredores: 4;
- Quantidade de seções por corredor: 4;
- Quantidade de itens por posição: 1;
- Quantidade total de itens: 16;
- Posição do ponto de expedição: (0,0).

4.1.2 GERAÇÃO DA LISTA DE PEDIDOS

A geração da lista de pedidos é realizada a partir de algumas restrições que foram definidas a fim de aproximar a análise ao ocorrido na prática, mas que também não provoque um nível de complexidade muito elevado ao algoritmo. Sendo assim, foi considerado o princípio de Pareto, onde aproximadamente 80% dos pedidos são direcionados a 20% dos itens

do armazém. O princípio também foi aplicado para determinação das quantidades de itens de cada pedido: com um limite de no máximo dois itens por pedido, 20% dos pedidos apresentam quantidade de itens igual a dois.

Restrições adotadas para a geração da lista de pedidos:

- ~80% dos pedidos para ~20% dos itens;
- Limite de no máximo 2 itens por pedido;
- ~20% dos pedidos com 2 itens

Após definidas as restrições, a variável relacionada à quantidade total de pedidos por lista de pedidos é estabelecida de modo a garantir simulações alternativas do tamanho da lista de pedidos. Como resultado, a lista é gerada contendo as informações de identificação dos pedidos, itens e quantidades.

Figura 3 - Bibliotecas para geração de dados

```
import pandas as pd
import numpy as np
```

Fonte: Figura do autor (2021)

Figura 4 - Criação das posições de armazenamento e layout do armazém

```
# Gerar os dados do wms que associa itens e localizações (coordenadas x e y).
#Df: Código item: inteiro
# Nome: o próprio índice
# Coord_X:[m]
# Coord_y:[m]

fileiras = 10
secoes = 10
prateleiras = 1
tamanho_armazem = fileiras*secoes*prateleiras

dict_ref_fileiras = {0:[2,1] , 1:[4,1], 2: [6,1], 3:[8,1], 4:[10,1], 5: [12,1], 6:[14,1], 7:[16,1], 8:[18,1], 9: [20,1]}
largura_prateleira = 1

ls_item = []
ls_nome = []
ls_fileira = []
ls_secao = []
ls_prateleira = []
ls_x = []
ls_y = []
```

Fonte: Figura do autor (2021)

Figura 5 - Geração dos itens do armazém

```

# gerando itens
num_item = -1
for f in range(fileiras):
    for s in range(secoes):
        for p in range(prateleiras):
            num_item = num_item + 1
            ls_item.append(num_item)
            ls_nome.append(num_item)
            ls_fileira.append(f)
            ls_secao.append(s)
            ls_prateleira.append(p)

            # calculo da coord x
            ls_x.append(dict_ref_fileiras[f][0])
            # calculo da coord y
            ls_y.append(dict_ref_fileiras[f][1] + s*largura_prateleira)

# salvando em um data frame
df_itens_loc = df = pd.DataFrame(list(zip(ls_item, ls_nome, ls_x, ls_y)),
                                  columns = ['Cod_Item', 'Nome', 'Coord_X', 'Coord_Y'])
print(df_itens_loc)

nome_file_csv = f"df_itens_{fileiras}_{secoes}_{prateleiras}_{tamanho_armazem}.csv"
df_itens_loc.to_csv(nome_file_csv)

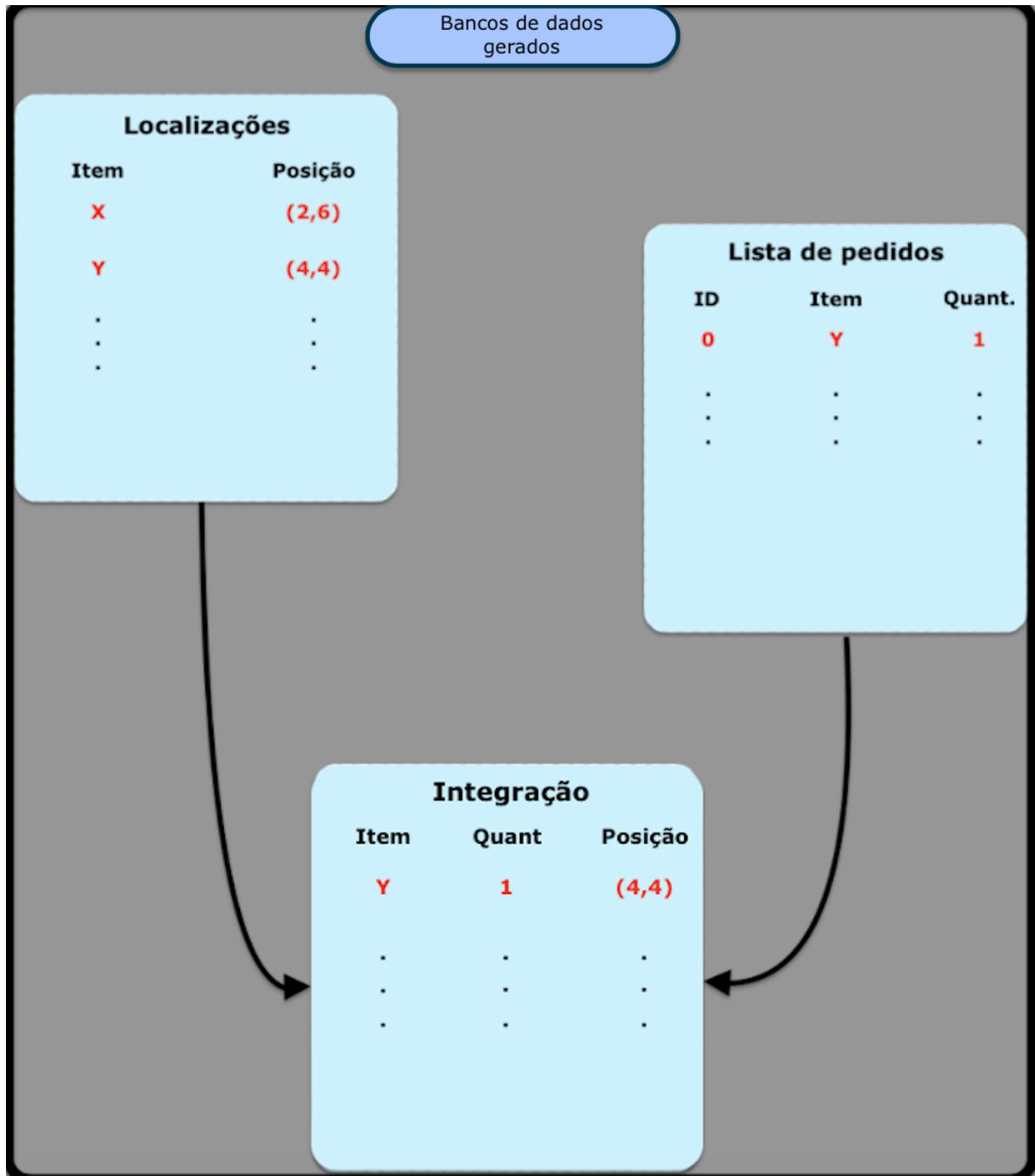
```

Fonte: Figura do autor (2021)

4.1.3 GERAÇÃO DE DADOS DE UM SISTEMA DE GERENCIAMENTO ARMAZÉNS

O sistema de gerenciamento do armazém é composto pelos dados gerados na definição do layout, onde estão contidas as informações sobre as posições dos itens, bem como a lista de pedidos. Dessa forma, é possível vincular os itens de um pedido a sua localização no armazém que será fundamental para a aplicação das ferramentas para análise das distâncias. Baseado no indicado pelos estudos de Saci (2020) a Figura 6 abaixo ilustra uma adaptação dos bancos de dados gerados como base para continuação das etapas.

Figura 6 – Bancos de dados com os atributos existentes em um WMS necessários ao trabalho

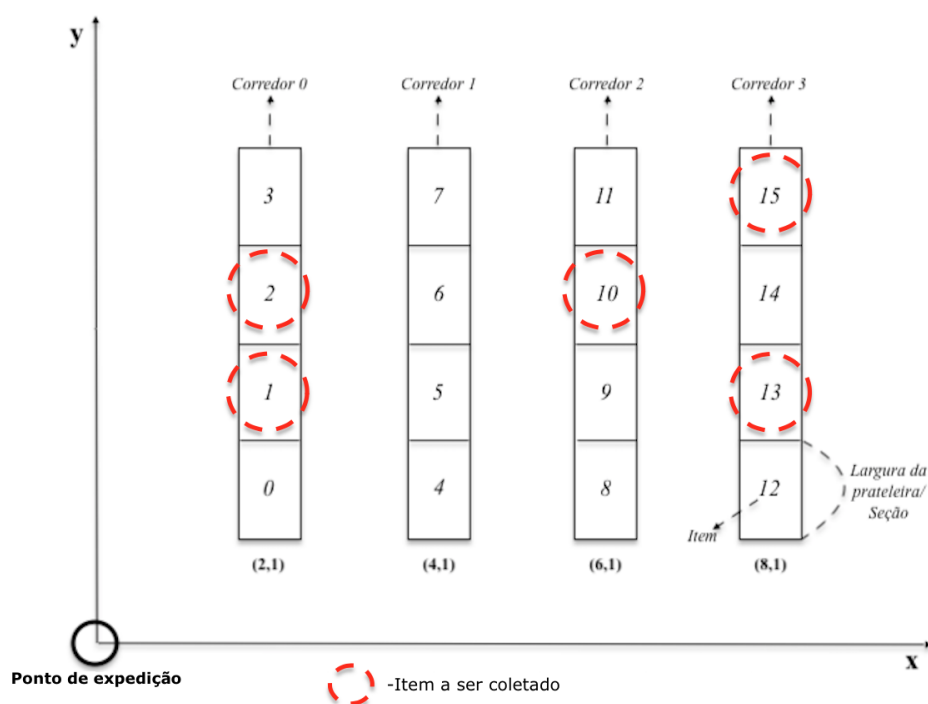


Fonte: Adaptado de Saci (2020)

4.2 IDENTIFICAÇÃO DE ESTRATÉGIAS DO SPRP

O problema consiste em determinar a menor rota que passa por cada local de armazenamento e retorna ao ponto de depósito/expedição, dada uma lista de locais de armazenamento no armazém e as distâncias entre cada par de locais.

Figura 7 - Exemplo de itens para coleta no armazém



Fonte: Adaptado de Saci (2020)

A Figura 7 acima demonstra uma situação em que cinco itens alocados em posições distintas no armazém devem ser coletados. Intuitivamente, se o operador coletar um item por vez e retornar ao ponto de expedição a distância total percorrida será elevada quando comparada a outras formas de realizar a coleta. Segundo Saci (2020) uma maneira de otimizar o *picking* é combinar os pedidos em uma mesma rota, essa estratégia é geralmente conhecida por *wave picking*. Dessa forma, uma das análises será baseada na aplicação de diferentes estratégias de

wave picking para uma mesma lista de pedidos com itens a serem coletados e seu impacto na distância total percorrida pelo responsável pela atividade de *picking*.

Outra forma que potencialmente aumentará a produtividade na atividade de *picking* é o agrupamento de itens do armazém por zonas/região para separação dos pedidos de modo a reduzir a distância total percorrida pelo operador na coleta ao mantê-lo em uma zona delimitada. Isso pode ser feito através da clusterização espacial também indicado no estudo de Saci (2020), que é a tarefa de agrupar um conjunto de pontos (localização de item) de forma que os objetos no mesmo *cluster* sejam mais semelhantes entre si do que os objetos em outros *clusters*. Esse agrupamento é delimitado a partir de uma variável que define a distância máxima caminhada pelo operador entre dois pontos de coleta. Dessa forma, o agrupamento dos pontos de coleta por clusters pode reduzir a distância percorrida pelo operador em cada rota de *picking*.

As duas técnicas recebem os dados disponíveis no banco de dados gerados na etapa anterior, onde estão contidas informações referentes às localizações de cada item para um modelo bidimensional do armazém e a lista de pedidos, com as quantidades e itens a serem coletados. A partir desses dados iniciais, a lista de pedidos é vinculada às posições(x,y) no modelo bidimensional para possibilitar o processamento das funções e cálculos associados a cada técnica identificada. A Tabela 1, resume as técnicas identificadas e suas respectivas variáveis de ajuste, que são alteradas ao longo das simulações a fim de se obter a melhor resposta (menor distância total).

Tabela 1 - Técnicas para análise de picking no armazém

Técnica	Variável de Ajuste
<i>Wave Picking</i>	Tamanho de onda
<i>Clustering</i> Espacial	Limiar de distância entre dois pontos

Fonte: Autor (2021)

4.3 ESTRATÉGIA DE *WAVE PICKING*

Para aplicar as análises de *wave picking* é necessário receber os dados simulados do banco de dados do WMS, criar funções, e então obter o resultado das rotas e distâncias percorridas ao coletar todos os itens contidos na lista de pedidos. A simulação consiste em

simular diferentes tamanhos de ondas e identificar o impacto provocado na distância total caminhada por um operador.

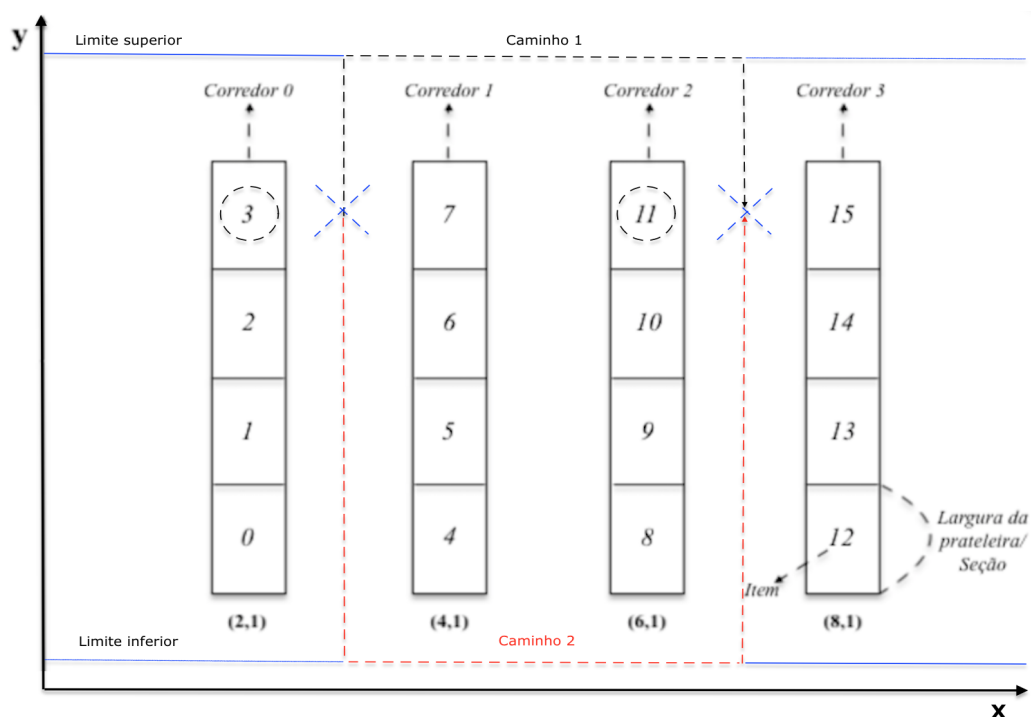
4.3.1 FUNÇÕES PARA APLICAÇÃO DA ANÁLISE DE *WAVE PICKING*

A partir dos dados extraídos do banco de dados com os atributos existentes em um WMS necessários ao trabalho, obtém-se quais itens devem ser coletados e onde estão suas posições no armazém. Agora, é necessário calcular a distância do ponto de partida até o item alvo, identificar o próximo item alvo, calcular sua distância e repetir os processamentos até passar por todos os itens restantes contidos na mesma onda de coleta e a distância de retorno ao ponto de expedição para se obter a distância total da caminhada de acordo com diferentes tamanhos de ondas, conforme Saci (2020) sugere.

4.3.1.1 Função para cálculo da mínima distância entre dois pontos

O cálculo da distância mínima entre dois pontos deve receber as coordenadas x e y do ponto inicial e do ponto final.

Figura 8 - Distância entre dois pontos



Fonte: Adaptado de Saci (2020)

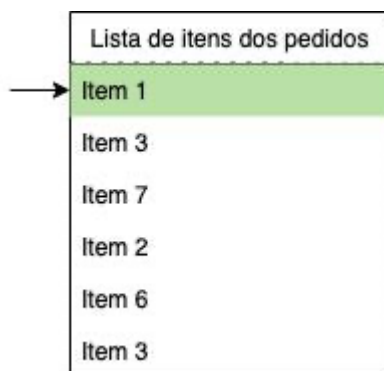
A organização física do armazém impede que o cálculo seja baseado na distância euclidiana, a função deve calcular a distância considerando a rota onde o operador pode transitar. Esse cálculo é realizado com base nos limites verticais do armazém, nas distâncias horizontais entre os corredores e na comparação entre o caminho por baixo com o caminho por cima do armazém.

No exemplo da Figura 8 acima, os limites estão tracejados de vermelho, as distâncias entre os corredores são de dois metros, a distância calculada é entre os itens três e onze, as alternativas de caminho estão tracejadas de azul e preto, o menor caminho é salvo e posteriormente utilizado no algoritmo em outras etapas/funções necessárias para a simulação.

4.3.1.2 Função para busca do próximo item a ser coletado da lista de pedidos

Essa função é utilizada para determinar qual o próximo ponto a ser coletado dentre os candidatos contidos na lista de pedidos para continuar a rota da onda de coleta.

Figura 9 - Exemplo de lista de itens dos pedidos



Fonte: Figura do autor (2021)

No exemplo indicado acima, o operador de *picking* se encontra na posição do item um e, a partir da função do cálculo de distância indicado no tópico anterior, as distâncias dos candidatos à próxima busca será processada. Sendo assim, a função irá processar as distâncias do ponto relacionado ao item um até a localização de cada item candidato restante dentro na lista, ou seja, a distância entre o item um e três, entre o item um e sete, um e dois, um e seis e,

um e três. Sequencialmente os resultados são comparados e a menor distância indicará qual item deve ser o próximo a ser coletado dentro da lista de pedidos.

4.3.1.3 Função para criação das ondas de coleta

Função para criação de *batches* de n pedidos a serem coletados. Essa função determina o tamanho das ondas de coletas e limita a função anterior ao tamanho de onda de coleta ao invés da lista inteira de pedidos, a Figura 10 abaixo ilustra uma representação da aplicação dessa função em conjunto com as outras considerando-se um tamanho de onda igual a três.

Figura 10 - Exemplo de lista de itens dos pedidos em batches

Lista de itens dos pedidos	ID da Onda
Item 1	1
Item 3	1
Item 7	1
Item 2	2
Item 6	2
Item 3	2

→ Tamanho de onda = 3

Fonte: Figura do autor (2021)

Sendo assim, as distâncias agora são processadas dentro de uma mesma onda, e a menor distância dentro do *batch* determinará qual será o próximo item a ser coletado.

4.3.1.4 Função para listagem da rota criada para coleta de onda

Função que registra as coordenadas associadas a cada item a ser coletado da maneira determinada pelas funções anteriores. Em resumo, essa função contém a rota que foi criada com cada item e sua respectiva posição (coordenadas x e y) para realização do *wave picking*.

Com esses dados é possível calcular a distância total do percurso do operador no armazém baseado no tamanho de onda de coleta. A Tabela 2 abaixo ilustra como é o formato

da resposta do algoritmo para uma onda de tamanho 3, considerando-se as coordenadas 0 em x e y como ponto de expedição.

Tabela 2 - Formato de resposta da análise 1

ID	Numero da Onda	Distancia da Rota[m]	Caminho Percorrido(x,y)	Pedidos/Onda
1	22	32	[[0, 0], (2, 1), (4, 4), (6, 9), [0, 0]]	3

Fonte: Figura do autor (2021)

4.3.2 SIMULAÇÃO DA ANÁLISE DE *WAVE PICKING*

Após configurar todas as funções necessárias para medir as distâncias na atividade de *picking* do armazém, é possível simular o impacto de diferentes rotas e ondas de coletas.

A abordagem para a simulação é bem simples, a lista de pedidos gerada no início do programa representa os pedidos em ordem cronológica (o topo sendo o primeiro pedido e o final da lista o último pedido), a separação por ondas é realizada em ordem cronológica também e, a partir da lógica da busca pelo item mais próximo como mostrado nos tópicos anteriores, a rota será criada.

Por fim, para simular o impacto da estratégia de separação por ondas na produtividade do armazém, o tamanho de onda será incrementado gradualmente e a distância total caminhada é obtida, permitindo observar o quanto da distância total pode ser reduzida ao incrementar a quantidade de pedidos por onda de coleta.

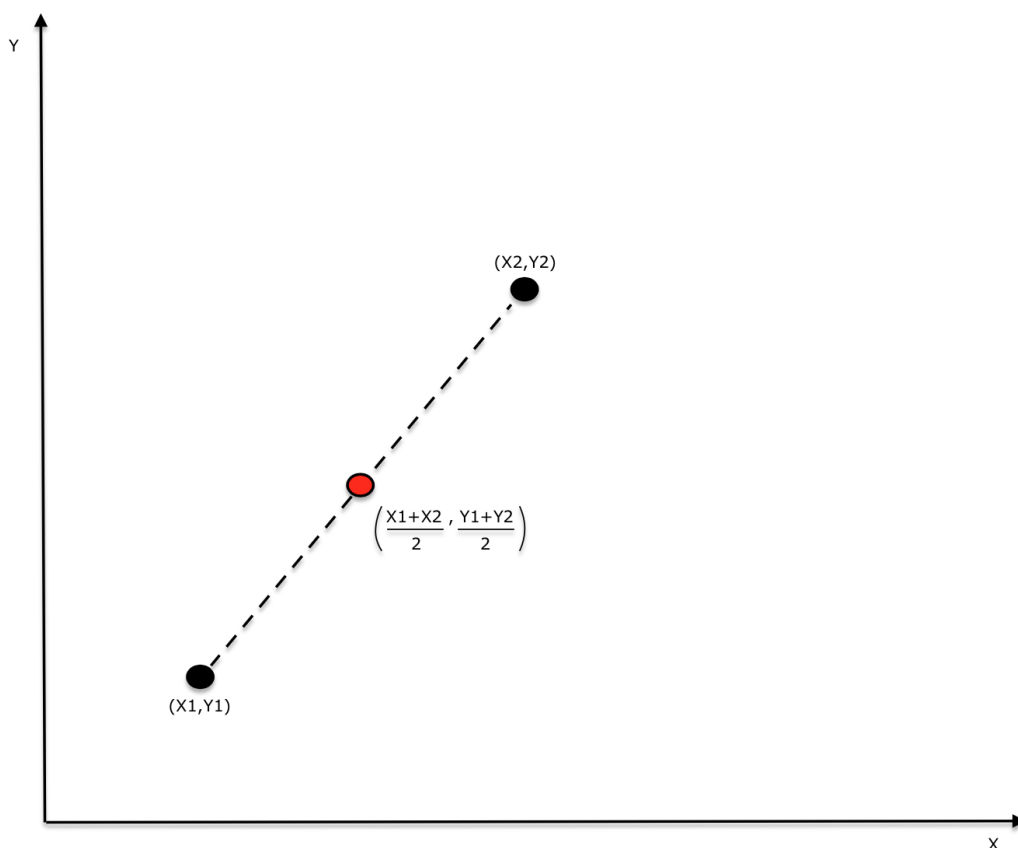
4.4 ESTRATÉGIA DE *CLUSTERING* ESPACIAL

Essa estratégia consiste em aplicar a *clusterização* espacial para determinar a separação dos pedidos que estarão nos *batches* com o objetivo de reduzir a distância total percorrida pelo operador responsável pela atividade de *picking*, visto que sua rota estará delimitada a uma zona com os pedidos de itens próximos uns aos outros, conforme sugerido por Saci (2020).

4.4.1 CONSIDERAÇÕES DOS PEDIDOS SINGULARES E PEDIDOS MÚLTIPLOS

Na geração de dados dos bancos de dados foi definido que um pedido pode conter um ou dois itens, o que significa que em um mesmo pedido o operador pode passar por até duas localizações distintas e, ao realizar a delimitação das zonas de coleta, pode ocorrer de um mesmo pedido apresentar localizações em zonas distintas. Dessa forma, é necessário separar os pedidos em singulares ou múltiplos e tratar os pedidos múltiplos de maneira diferente para evitar de encontrar dois *clusters* em um mesmo *batch*. Para resolver essa situação, as localizações dos itens a serem coletados dos pedidos múltiplos são identificadas e o ponto médio é calculado entre os dois pontos, que será o ponto considerado para determinação dos *clusters*.

Figura 11 - Distância média entre dois pontos



Fonte: Adaptado de Saci (2020)

A Figura 11 acima indica um pedido múltiplo com as localizações dos itens representadas pelos pontos pretos nas posições X1, Y1 e X2, Y2, porém o ponto que será considerado para realização dos cálculos e determinação do *cluster* será o ponto médio entre os dois pontos, que na Figura 11 está representado pelo ponto vermelho.

Após o cálculo do ponto médio para os pedidos múltiplos, os *clusters* para os pedidos múltiplos são determinados da mesma maneira em que os pedidos singulares são determinados.

4.4.2 CONSIDERAÇÕES PARA CÁLCULO DAS DISTÂNCIAS

Segundo Saci (2020), o método convencional de se calcular a distância, através da distância euclidiana, não é adequado, a distância euclidiana indicaria a movimentação em linha reta de um ponto ao outro, mas a distância percorrida pelo operador ao caminhar pelo armazém deve considerar a organização física do espaço e por onde o operador pode ou não transitar, como mostrado também na primeira estratégia de *wave picking*. Sendo assim, a mesma função determinada na estratégia anterior, será aproveitada para realizar os cálculos relacionados às distâncias nessa estratégia, extremamente importante para a determinação dos *clusters*.

4.4.3 CLUSTERIZAÇÃO

A lista de pedidos é recebida com os pedidos singulares e múltiplos distribuídos de forma aleatória e em seguida são separados em duas partes para realização da *clusterização*: pedidos únicos/singulares e pedidos múltiplos.

Os pedidos singulares são diretamente processados e os *clusters* são obtidos. Os *clusters* para localizações de itens são determinados através da biblioteca **scipy** e a função **fcluster**, utilizando-se a função personalizada para o cálculo da distância caminhada no armazém ao invés da distância euclidiana, ou seja, os itens que estão dentro de um limiar de distância pré-definido ao calcular a distância pela função personalizada de busca, pertencerão ao mesmo *cluster*.

O código em python na Figura 12 abaixo, indica como é a função para criação dos *clusters*, e a função personalizada para cálculo da distância, é nomeada como “*distancia_de_busca1*”, indicada na Figura 13 e em seguida sua explicação.

Figura 12 - Função para determinação de clusters

```
#Criar clusters baseado no limite de distancia
from scipy.spatial.distance import pdist
from scipy.cluster.hierarchy import ward, fcluster

def cluster_locations(lista_coord, limite_dist):
    ##Distancia caminhada
    dc = pdist(lista_coord, distancia_de_busca1)
    Y = ward(dc)
    fclust1 = fcluster(Y, t = limite_dist, criterion = 'distance')

    return fclust1
```

Fonte: Adaptado de Saci (2020)

Essa função, também mostrada no trabalho de Saci (2020), foi adaptada para realizar a clusterização a partir da função criada abaixo que foi criada baseada na distância onde pode ocorrer fluxo de movimentação e respeitando as limitações do armazém definido para esse trabalho.

Figura 13 - Função para determinação da distância caminhada

```

def distancia_de_buscal(u, v):
    #INCLUIR LIMITES DO ARMAZÉM

    y_limite_inferior, y_limite_superior = 0.5, 11
    # Start Point
    x1, y1 = u[0], u[1]
    # End Point
    x2, y2 = v[0], v[1]
    # Distancia eixo x
    distancia_x = abs(x2 - x1)
    # Distancia eixo y
    if x1 == x2:
        distancia_y1 = abs(y2 - y1)
        distancia_y2 = distancia_y1
    else:
        distancia_y1 = (y_limite_superior - y1) + (y_limite_superior - y2)
        distancia_y2 = (y1 - y_limite_inferior) + (y2 - y_limite_inferior)
    # Distância mínima no eixo y
    distancia_y = min(distancia_y1, distancia_y2)
    # Distancia total
    distancia = distancia_x + distancia_y

    return distancia

```

Fonte: Adaptado de Saci (2020)

Na Figura 13 acima, é considerado um armazém onde o operador pode caminhar no mínimo até a posição de meio metro e no máximo onze metros na vertical. Esses limites são definidos para indicar a rota de passagem por cima e por baixo dos corredores, por onde o operador passa ao se locomover de um corredor para outro no armazém e, são utilizados na função para comparar se é melhor se locomover por cima ou por baixo dos corredores ao se direcionar ao ponto alvo. Já as distâncias horizontais são calculadas diretamente, não existe comparação de duas possibilidades, a distância é determinada pelo módulo da subtração da coordenada horizontal do corredor alvo pela coordenada horizontal do corredor de origem.

Para melhor exemplificar, considera-se dois pontos(u,v) de argumentos da função como sendo a posição de origem(u) o item um nas coordenadas (2,2) e, a posição alvo(v), o item sete nas coordenadas (4,4):

$$x1 = 2; y1 = 2;$$

$$x_2 = 4; y_2 = 4;$$

$$\text{Distância em x: } |4 - 2| = \mathbf{2[m]}$$

Se estiverem no mesmo corredor ($x_1=x_2$), só existe um caminho vertical. Caso contrário (caso do exemplo), são consideradas duas opções possíveis:

$$\text{Opção 1: Limite superior} - y_1 + \text{Limite superior} - y_2 \quad (1)$$

$$(11 - 2) + (11 - 4) = \mathbf{16[m]}$$

$$\text{Opção 2: } (y_1 - \text{Limite inferior}) + (y_2 - \text{Limite inferior}) \quad (2)$$

$$(2 - 0.5) + (4 - 0.5) = \mathbf{5[m]}$$

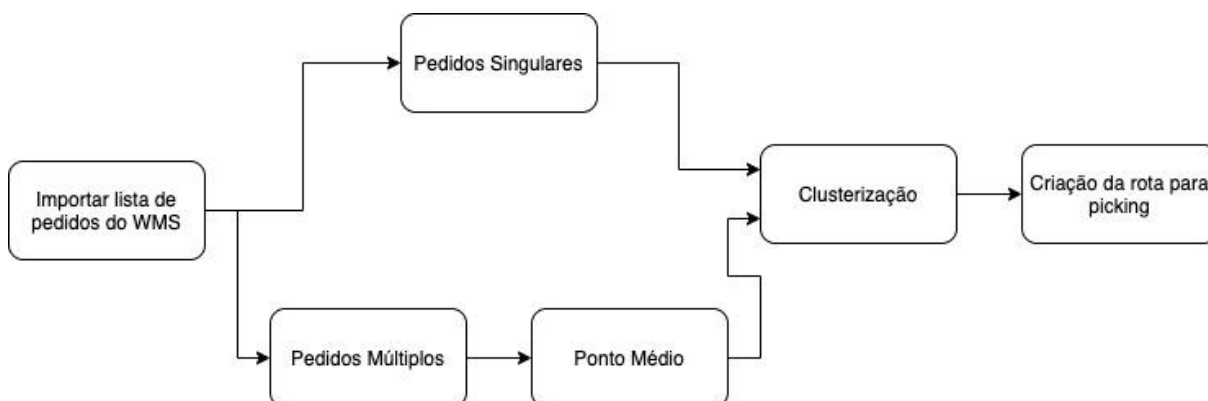
Ou seja, nesse cenário, ao se comparar as alternativas da locomoção por cima ou por baixo, a segunda opção, de transitar por baixo do armazém, resulta em uma distância menor a ser percorrida e, portanto, será considerada para o cálculo da distância:

$$\text{Distância} = \text{Distância em y} + \text{Distância em x}$$

$$\text{Distância} = 5 + 2 = \mathbf{7[m]}$$

De maneira simplificada, a Figura 14 abaixo resume como os pedidos são processados até a criação da rota a partir da estratégia de clusterização para o operador do armazém realizar a coleta.

Figura 14 - Modelo simplificado da segunda estratégia



Fonte: Adaptado de Saco (2020)

Após separar os pedidos em *clusters*, algumas funções que não estão especificadas no modelo simplificado são necessárias até chegar na etapa da criação da rota de coleta dos itens com o mapeamento indicando as localizações que o operador deve se direcionar, como o preenchimento de itens a serem coletados até atingir o tamanho de onda pré-definido, a função de comparar as distâncias dos itens candidatos a fim de selecionar o menos distante e, por fim, a simulação para diferentes limiares de distância para criação de *clusters* bem como tamanhos de ondas.

4 SIMULAÇÃO DA ANÁLISE DE CLUSTERIZAÇÃO ESPACIAL

Os *clusters* são diretamente impactados pelo valor definido para o limiar da distância, visto que serve como referência para realização dos agrupamentos. Ao definir um limiar de três metros, a função busca agrupar itens dentro dessa distância (a partir da função personalizada) e, conseqüentemente, se o armazém tiver um espaço relativamente grande, vários clusters serão criados. Por outro lado, ao definir o limiar em cinquenta metros, para o mesmo armazém, poucos clusters serão criados. Sendo assim, é necessário simular diversos valores de limiar de distância para se obter o valor que resulta na menor distância total.

Para isso, uma função é criada para simular a distância total percorrida a partir de um limiar mínimo e esse valor é incrementado em um metro por vez e sua resposta obtida até um valor máximo de limiar de distância, para comparação e ajuste de limiar da distância para *clusterização*.

5 RESULTADOS

Para realização da análise comparativa, dois layouts de armazéns são considerados. Para cada layout, as mesmas técnicas são aplicadas, os resultados obtidos e analisados.

5.1 CENÁRIOS PARA SIMULAÇÃO

A simulação é realizada em dois layouts de armazéns, ambos com a mesma quantidade de itens, mesmas demandas (lista de pedidos) e variações de demanda, porém, o segundo layout consiste no rearranjo dos mesmos itens em metade do espaço do primeiro layout.

5.1.1 Geração dos itens e suas localizações no cenário 1

A primeira simulação é feita em cima de um layout com as seguintes características:

- Número de corredores: 10
- Número de seções: 10
- Número de prateleiras: 1
- Quantidade de itens: 100

As posições alocadas a cada item representam onde o operador deve estar posicionado para realização da coleta do respectivo item. Essas posições servem de base para posteriormente realizar os cálculos das distâncias percorridas pelo operador.

Para a geração da lista de pedidos foi criado um código na linguagem de programação python com a importação das bibliotecas pandas e numpy e o código foi elaborado como mostrado nas figuras abaixo.

Figura 15 - Bibliotecas para geração de dados

```
import pandas as pd
import numpy as np
```

Fonte: Figura do autor (2021)

Figura 16 - Criação das posições de armazenamento e layout do armazém

```

# Gerar os dados do wms que associa itens e localizações (coordenadas x e y).

#Df: Código item: inteiro
# Nome: o próprio índice
# Coord_X:[m]
# Coord_y:[m]

fileiras = 10
secoes = 10
prateleiras = 1
tamanho_armazem = fileiras*secoes*prateleiras

dict_ref_fileiras = {0:[2,1], 1:[4,1], 2:[6,1], 3:[8,1], 4:[10,1], 5:[12,1], 6:[14,1], 7:[16,1], 8:[18,1], 9:[20,1]}
largura_prateleira = 1

ls_item=[]
ls_nome=[]
ls_fileira=[]
ls_secao=[]
ls_prateleira=[]
ls_x=[]
ls_y=[]

```

Fonte: Figura do autor (2021)

A partir da Figura 16 acima no dicionário “dict_ref_fileiras”, é possível observar o distanciamento entre cada corredor de dois metros, no qual cada item é associado a posição (x,y) com um incremento de duas unidades na coordenada x no próximo item. Ou seja, a distância total na horizontal pode chegar a 20 metros.

Figura 17 - Geração dos itens do armazém

```

# gerando itens
num_item = -1
for f in range(fileiras):
    for s in range(secoes):
        for p in range(prateleiras):
            num_item = num_item + 1
            ls_item.append(num_item)
            ls_nome.append(num_item)
            ls_fileira.append(f)
            ls_secao.append(s)
            ls_prateleira.append(p)

            # calculo da coord x
            ls_x.append(dict_ref_fileiras[f][0])
            # calculo da coord y
            ls_y.append(dict_ref_fileiras[f][1] + s*largura_prateleira)

# salvando em um data frame
df_itens_loc = df = pd.DataFrame(list(zip(ls_item, ls_nome, ls_x, ls_y)),
                                columns=['Cod_Item', 'Nome', 'Coord_X', 'Coord_Y'])
print(df_itens_loc)

nome_file_csv = f"df_itens_{fileiras}_{secoes}_{prateleiras}_{tamanho_armazem}.csv"
df_itens_loc.to_csv(nome_file_csv)

```

Fonte: Figura do autor (2021)

Como resultado dessa etapa, um arquivo *csv* contendo as informações sobre o ID, nome e posição dos itens contidos no armazém foi gerado, como exemplificado na Figura 18 abaixo.

Figura 18 - Resumo do dataframe do Layout 1.

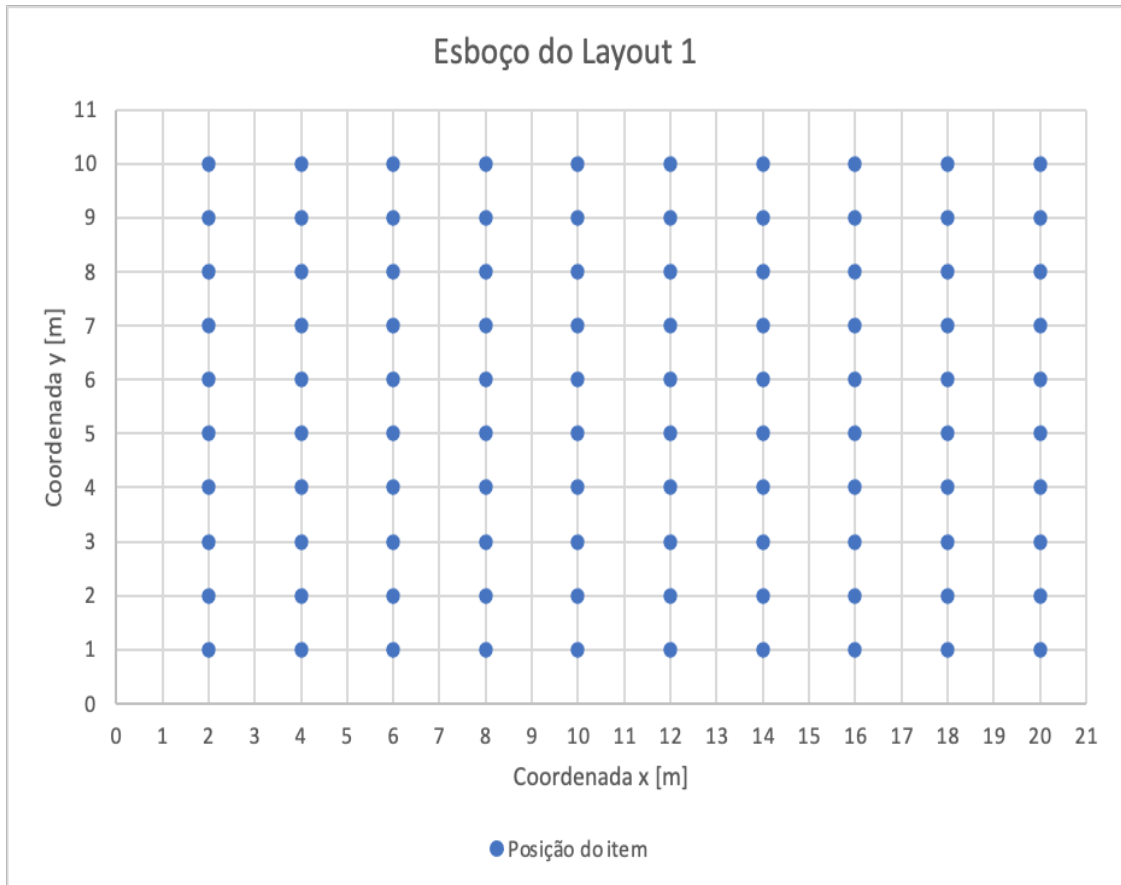
	Cod_Item	Nome	Coord_X	Coord_Y
0	0	0	2	1
1	1	1	2	2
2	2	2	2	3
3	3	3	2	4
4	4	4	2	5
..
95	95	95	20	6
96	96	96	20	7
97	97	97	20	8
98	98	98	20	9
99	99	99	20	10

[100 rows x 4 columns]

Fonte: Autor (2021)

Para melhor visualizar os dados gerados, a Figura 19 abaixo ilustra um esboço do Layout 1 com as posições alocadas aos itens definidos para essa simulação.

Figura 19 - Esboço do Layout 1



Fonte: Adaptado do Saci (2021)

5.1.2 Geração da lista de pedidos do cenário 1

A geração da lista de pedidos será baseada em algumas restrições que foram definidas a fim de aproximar a simulação ao que ocorre na prática, mas que também não provoque um nível de complexidade muito elevado ao algoritmo. Sendo assim, foi considerado o princípio de Pareto, onde aproximadamente 80% dos pedidos são direcionados a 20% dos itens do armazém. O princípio também foi aplicado para determinação das quantidades de itens de cada pedido: com um limite de no máximo 2 itens por pedido, 20% dos pedidos apresentam quantidade de itens igual a 2.

Restrições adotadas para a geração da lista de pedidos:

- ~80% dos pedidos para ~20% dos itens;
- Limite de no máximo 2 itens por pedido;

- ~20% dos pedidos com 2 itens

Em seguida, é necessário simular a demanda a partir da quantidade total de pedidos por lista de pedidos. Da mesma maneira como ocorreu na etapa de geração e alocação dos itens, o código foi elaborado na linguagem de programação python, com a importação das bibliotecas pandas e numpy, e o código é mostrado nas figuras abaixo.

Figura 20 - Definição das restrições da lista de pedidos

```
# inicia o gerador de números aleatórios para escolha dos
# produtos com 80% da procura
seed_1 = 1234567

rng_top_produtos = np.random.default_rng(seed_1)

# escolher aleatoriamente 20% dos itens
porcentagem_top_produtos = 0.2
size_top_produtos = int(porcentagem_top_produtos*tamanho_armazem)
top_produtos = rng_top_produtos.choice(tamanho_armazem, size_top_produtos, replace=False)

not_top_produtos = []
for p in range(tamanho_armazem):
    if p not in top_produtos:
        not_top_produtos.append(p)

# inicia o gerador de números aleatórios
# gerar pedidos(ordens de compra)
seed_2 = 45678912
seed_3 = 678923
#criação do que estará no dataframe
ls_pedido = []
ls_itens = []
ls_quant = []

rng_itens = np.random.default_rng(seed_2)
rng_pedidos = np.random.default_rng(seed_3)

total_pedidos = 100
num_pedido = 0
```

Fonte: Autor (2021)

Figura 21 - Geração da lista de pedidos

```

while num_pedido < total_pedidos:
    # valor aleatório que determina se item é top
    alpha = rng_itens.random()

    # valor aleatório que determina a quantidade to pedido
    beta = rng_pedidos.random()

    # 20% dos pedidos têm quantidade = 2
    quant_itens_pedido = 1
    if beta < 0.2:
        quant_itens_pedido = 2

    # gera os itens
    for p in range(quant_itens_pedido):
        if alpha < 0.8:
            item = rng_itens.choice(top_produtos, 1)
        else:
            item = rng_itens.choice(not_top_produtos, 1)

    #lista produtos
    ls_pedido.append(num_pedido)
    ls_itens.append(item[0])
    ls_quant.append(quant_itens_pedido)

    num_pedido=num_pedido + 1

df_pedidos = pd.DataFrame(list(zip(ls_pedido, ls_itens,ls_quant)),
                           columns =['ID_pedido', 'Item', 'Quant_itens'])

print(df_pedidos)
nome_file_csv = f"df_ordens_de_compra_{total_pedidos}.csv"
df_pedidos.to_csv(nome_file_csv)

```

Fonte: Autor (2021)

Para a geração de dados, foi determinada uma lista de pedidos de 100 ordens e, em resumo, a lista saiu conforme a imagem abaixo.

Figura 22 - Resumo da lista de pedidos da análise 1

	ID_pedido	Item	Quant_itens
0	0	7	2
1	0	96	2
2	1	1	1
3	2	10	1
4	3	14	1
..
114	96	81	1
115	97	67	1
116	98	65	2
117	98	10	2
118	99	96	1

[119 rows x 3 columns]

Fonte: Autor (2021)

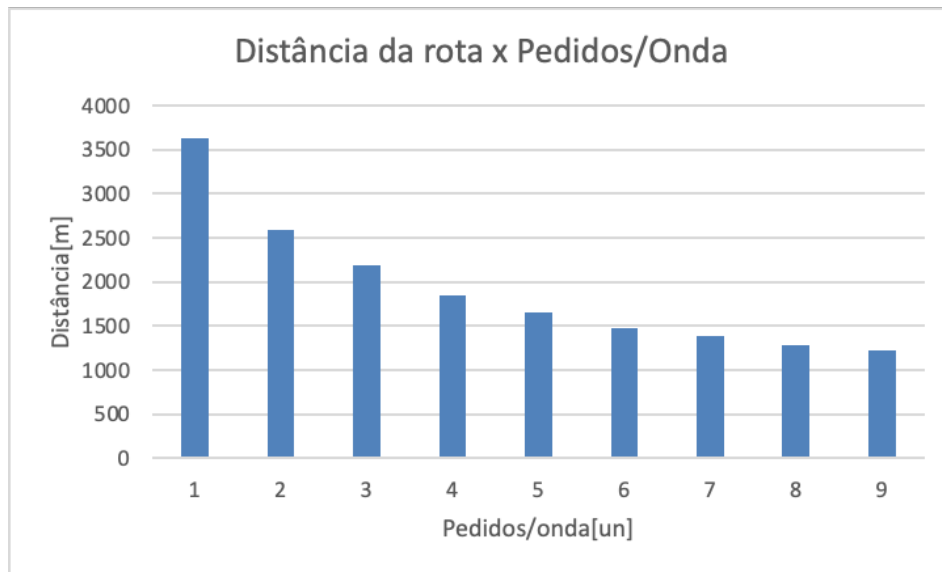
A geração da lista de pedidos resultou em 119 linhas por causa dos pedidos com dois itens, cada pedido com dois itens gerou duas linhas na lista de pedidos indicando o respectivo item, mas mantendo o ID do pedido.

5.1.3 Primeira análise do cenário 1

A partir da lista de pedidos gerada, a primeira análise consiste em simular diferentes tamanhos de onda e avaliar o impacto na redução da distância total na atividade de *picking* realizada por um operador. O programa foi executado com a lista de pedidos determinada no tópico anterior e o tamanho de onda foi incrementado de um em um até o tamanho máximo de nove pedidos por onda de coleta. É considerado também um ponto de expedição na posição (0,0), no qual o operador começa e termina cada rota de coleta.

O resultado é demonstrado na Figura 23 abaixo.

Figura 23 - Resultado em colunas da primeira análise do Layout 1



Fonte: Adaptado de Saci (2020)

Tabela 3 - Resultado numérico da primeira análise do Layout 1

Pedidos/Onda[un]	Distância da rota[m]
1	3626
2	2594
3	2190
4	1854
5	1660
6	1478
7	1382
8	1284
9	1226

Fonte: Autor (2021)

Ao separar pedidos em ondas de coleta, a distância é reduzida significativamente, enquanto uma busca sem agrupamentos resulta em uma distância de 3626 metros, a melhor opção de agrupamento (9 pedidos por onda), apresenta uma distância de apenas 1226 metros, ou seja, aproximadamente 33,8% da distância que seria percorrida sem agrupamento.

O resultado detalhado das rotas obtidas a partir da separação dos pedidos em grupos de nove é demonstrado na Tabela 4 abaixo.

Tabela 4 - Resultado detalhado da primeira análise do Layout 1

N° da Onda	Distância da Rota	Caminho Percorrido	Pedidos/Onda
0	92	[[0, 0], (2, 2), (4, 1), (4, 5), (6, 10), (8, 10), (10, 7), (12, 8), (20, 7), (2, 8), [0, 0]]	9
1	82	[[0, 0], (4, 6), (6, 10), (6, 8), (12, 8), (14, 8), (14, 4), (14, 3), (20, 4), [0, 0]]	9
2	88	[[0, 0], (2, 1), (4, 1), (2, 6), (10, 3), (10, 4), (14, 6), (20, 4), [0, 0]]	9
3	88	[[0, 0], (4, 1), (2, 6), (6, 8), (14, 6), (16, 2), (16, 7), (18, 2), [0, 0]]	9
4	90	[[0, 0], (2, 1), (4, 1), (2, 10), (10, 7), (14, 3), (20, 1), (20, 6), (20, 7), [0, 0]]	9
5	84	[[0, 0], (4, 1), (2, 6), (6, 8), (12, 8), (10, 3), (18, 2), [0, 0]]	9
6	88	[[0, 0], (2, 1), (4, 2), (6, 1), (10, 7), (2, 10), (2, 8), (18, 7), (18, 2), [0, 0]]	9
7	96	[[0, 0], (2, 1), (4, 4), (2, 10), (6, 7), (12, 8), (16, 7), (16, 2), (18, 8), [0, 0]]	9
8	114	[[0, 0], (6, 1), (6, 8), (8, 9), (2, 10), (2, 6), (12, 1), (10, 4), (14, 6), (20, 7), [0, 0]]	9
9	96	[[0, 0], (2, 1), (2, 6), (2, 8), (4, 5), (10, 7), (14, 6), (14, 3), (18, 2), (16, 7), [0, 0]]	9
10	86	[[0, 0], (10, 3), (14, 6), (16, 7), (18, 2), (20, 1), (20, 7), [0, 0]]	9
11	78	[[0, 0], (4, 1), (2, 6), (2, 8), (8, 5), (10, 2), (10, 3), (10, 7), (14, 8), [0, 0]]	9
12	88	[[0, 0], (2, 1), (6, 8), (10, 7), (12, 8), (14, 8), (14, 6), (16, 7), (18, 2), [0, 0]]	9
13	56	[[0, 0], (4, 1), (20, 7), [0, 0]]	9

Fonte: Autor (2021)

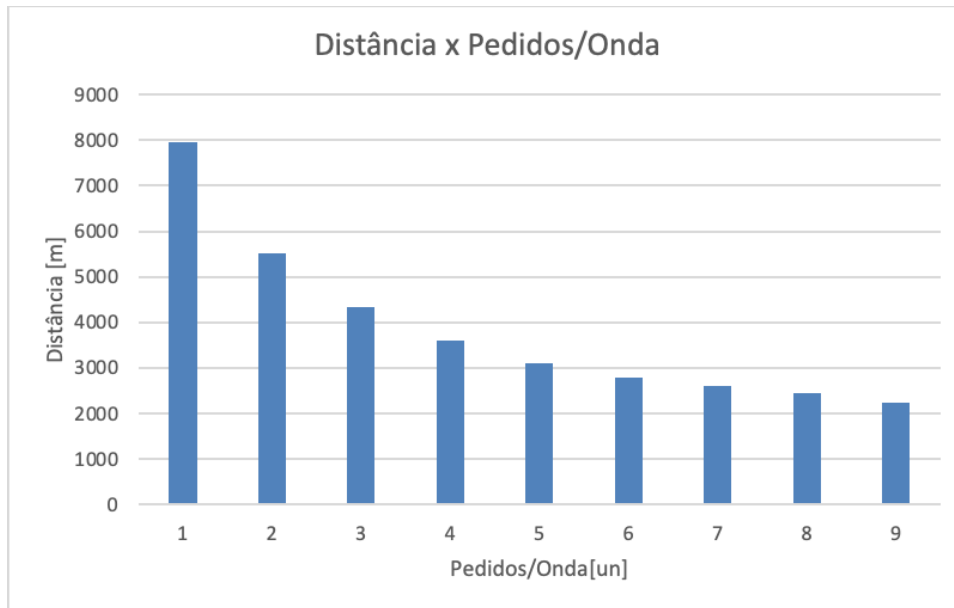
Alguns caminhos não indicam exatamente 9 pontos de coleta, como por exemplo, a onda de número 10 indica 6 pontos de coleta, isso se deve ao fato dessa onda conter itens repetidos, mas ainda são coletados nove itens por pedidos. E, a última onda (onda treze) é preenchida com os itens restantes da lista de pedidos, no caso dessa análise, restaram apenas dois pontos de coleta.

5.1.3.1 Primeira análise do cenário 1 com demanda alterada

A simulação foi novamente realizada para observar a diferença do comportamento ao alterar o tamanho da lista de pedidos para uma lista maior, o que significa simular uma demanda mais elevada. Foi considerado então um tamanho de 200 pedidos nessa nova lista, no qual, considerando-se os pedidos com mais de um item a ser coletado, o total de itens a serem buscados resultou em 234.

O resultado da primeira análise para esse cenário com a demanda aumentada, foi como mostrado abaixo:

Figura 24 - Resultado da primeira análise do Layout 1 com demanda alterada



Fonte: Adaptado de Saci (2020)

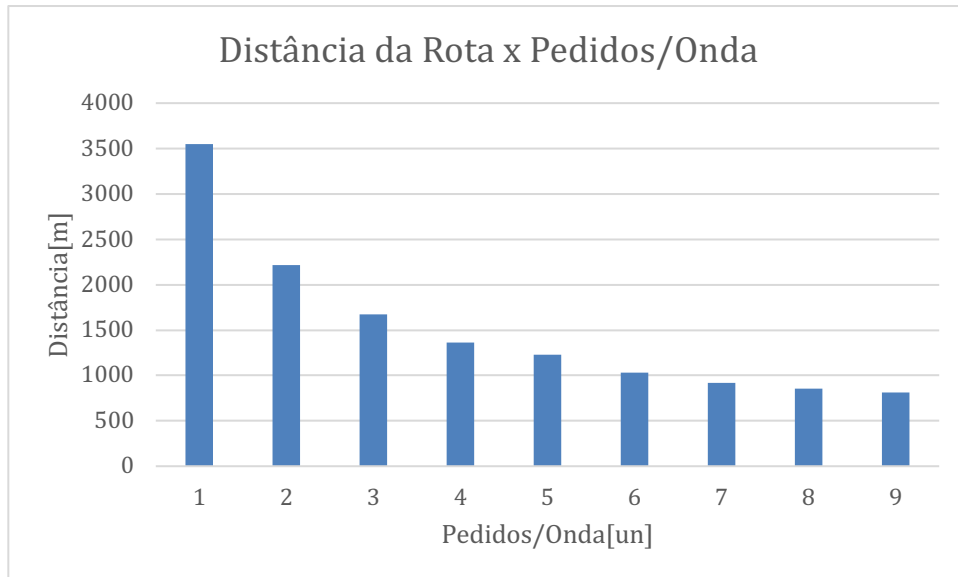
O agrupamento de nove itens por busca resultou em uma distância de 2228 metros, o que novamente indica que a melhor maneira de fazer a coleta de itens na estratégia de *wave picking* é criando batches de nove itens por busca.

5.1.4 Segunda análise do cenário 1

A segunda análise do primeiro cenário consiste em aplicar a estratégia de *clustering* espacial para determinar a separação dos pedidos, utilizando-se a mesma lista de pedidos da primeira análise, bem como o layout.

Ao reorganizar os pedidos por *clusters* baseado em um limiar de distância inicial de vinte metros e, aplicando-se a mesma simulação da etapa anterior, os resultados são obtidos conforme a Figura 25 abaixo.

Figura 25 - Resultado em colunas da segunda análise do Layout 1



Fonte: Adaptado de Saci (2020)

Tabela 5 - Resultado numérico da segunda análise do Layout 1

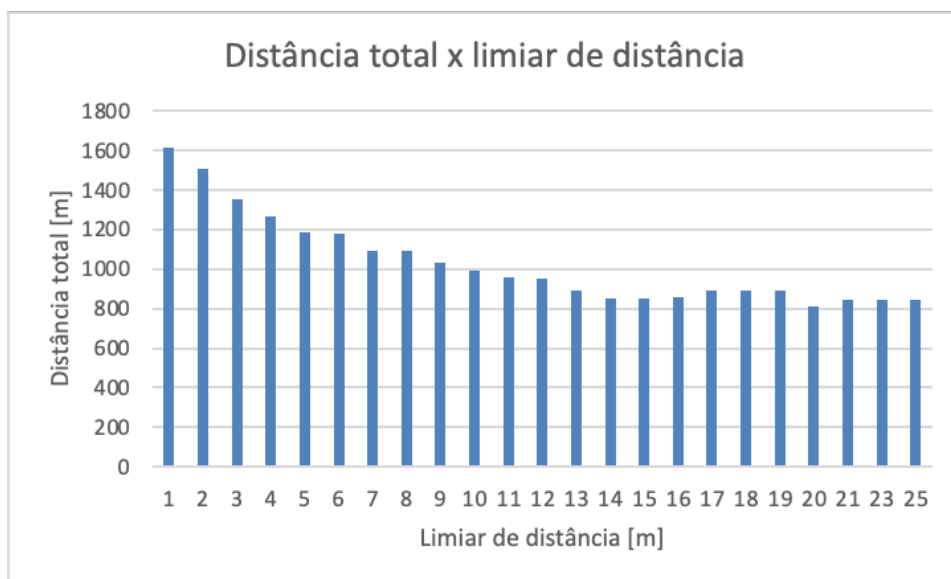
Pedidos/Onda[un]	Distância da rota[m]
1	3546
2	2194
3	1706
4	1316
5	1208
6	1056
7	940
8	812
9	812

Fonte: Autor (2021)

Os resultados indicam que para um limiar inicial de vinte metros na criação dos clusters, oito pedidos por onda de coleta garante um bom resultado. Sendo assim, a simulação para ajuste do limiar será fixada em ondas de oito pedidos para coleta.

O resultado do impacto da distância total percorrida a partir da variação do limiar de distância para criação dos *clusters* é indicado na Figura 26 abaixo.

Figura 26 - Ajuste de limiar de distância da segunda análise do Layout 1



Fonte: Adaptado de Saci (2020)

A partir da análise é possível verificar que a variação no limiar da distância para criação dos *clusters* pode reduzir significativamente a distância total percorrida pelo operador.

Ao se comparar as situações extremas de ajuste de limiar, a resposta é obtida conforme Tabela 6 abaixo:

Tabela 6 - Dados do ajuste de limiar de distância do Layout 1

Limiar da distância[m]	Distância total percorrida[m]	N° de Clusters[un]	Tamanho de Onda
1	1614	16	8
20	812	5	8

Fonte: Autor (2021)

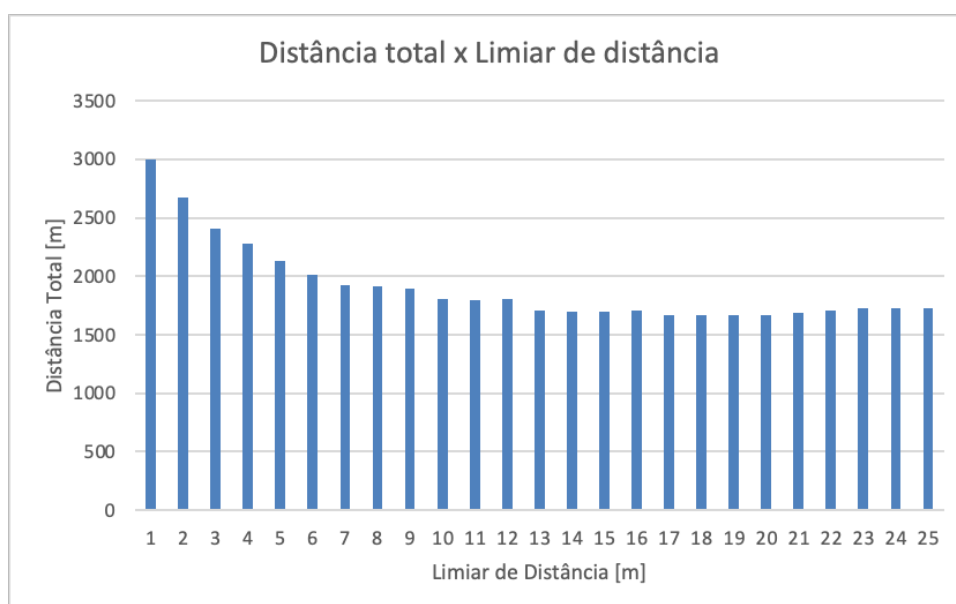
Conforme esperado, quanto menor o limiar, maior a quantidade de *clusters* e quanto maior o limiar, menor a quantidade de *clusters*. Ao criar muitas zonas no armazém, as ondas de coletas dificilmente serão preenchidas com o tamanho esperado de pedidos. Isso acontece devido ao formato do modelo proposto, no qual a onda é preenchida somente por itens da mesma zona. E, como cada zona dispõe de poucos itens candidatos à coleta, as chances de fechar uma onda de coleta antes do número esperado são maiores, isso significa que o operador deve retornar ao ponto de expedição mais vezes e, conseqüentemente, aumentar sua distância total percorrida. Por outro lado, ao criar poucas zonas no armazém, o modelo realiza as separações com pouca diferenciação entre os itens, até o ponto onde existe somente uma zona que seria o armazém inteiro.

Na análise acima, os valores de limiar de distância foram testados e foi possível encontrar uma redução de até 49,7%, entre o limiar equivalente a 1 metro e 20 metros.

5.1.4.1 Segunda análise do cenário 1 com demanda alterada

A mesma simulação foi aplicada para uma lista de tamanho duas vezes maior e o resultado obtido foi o seguinte:

Figura 27 - Ajuste de limiar de distância da segunda análise do Layout 1 com demanda alterada



Fonte: Autor (2021)

Nessa segunda análise, o ajuste de limiar demonstrou que a menor distância é obtida quando o ajuste está em 17 metros, resultando em uma distância total de 1666 metros, foi possível encontrar uma redução de até aproximadamente 44,4%, entre o limiar equivalente a 1 metro e 20 metros.

5.1.5 Comparação das Análises do Cenário 1

A simulação para o primeiro cenário com as duas estratégias identificadas proporcionou impactos significativos na distância total das rotas do operador, responsável pela atividade de *picking*, no qual pode representar um aumento de produtividade no armazém como um todo.

Para analisar os impactos obtidos a partir das estratégias aplicadas no primeiro cenário na movimentação do operador, três situações são consideradas:

- Sem aplicação de estratégias, em outras palavras, coleta orientada pela chegada cronológica dos pedidos, sem separação por *batches*;
- Melhor resposta obtida a partir da aplicação da estratégia de *wave picking*;
- Melhor resposta obtida a partir da aplicação da estratégia de clusterização espacial.

Figura 28 - Resultados das estratégias no Layout 1

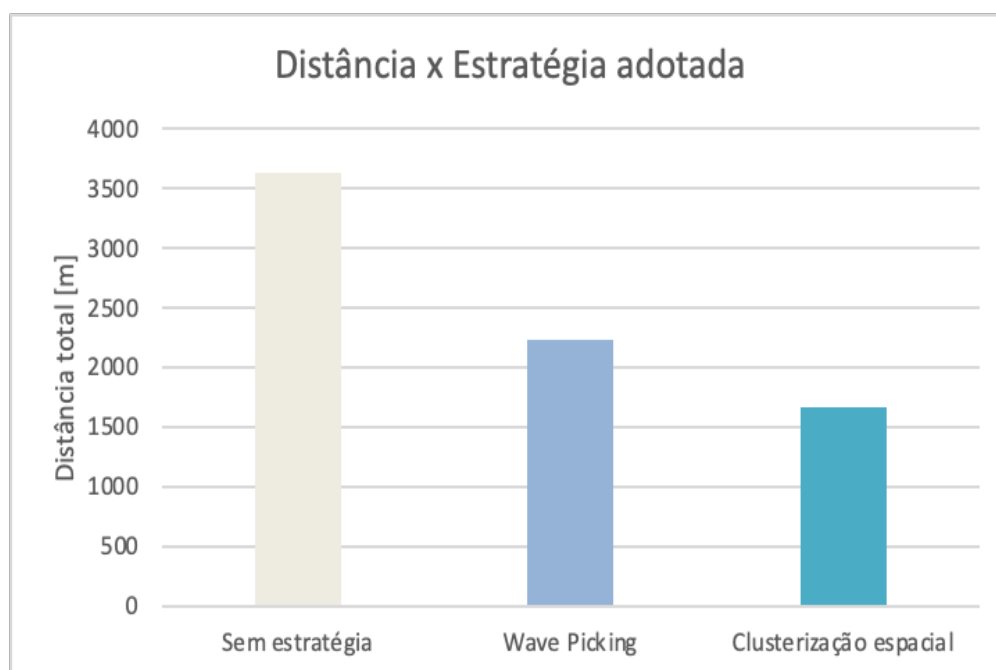


Fonte: Autor (2021)

Os resultados indicam uma redução de 66,2% na distância ao aplicar a estratégia de wave picking e 77,6% com a estratégia de clusterização espacial, ao se comparar com a distância obtida sem aplicar estratégias de movimentação.

Na análise que considera a lista de pedidos duplicada, o resultado foi dado da seguinte forma:

Figura 29 - Resultados das estratégias no Layout 1 com demanda alterada



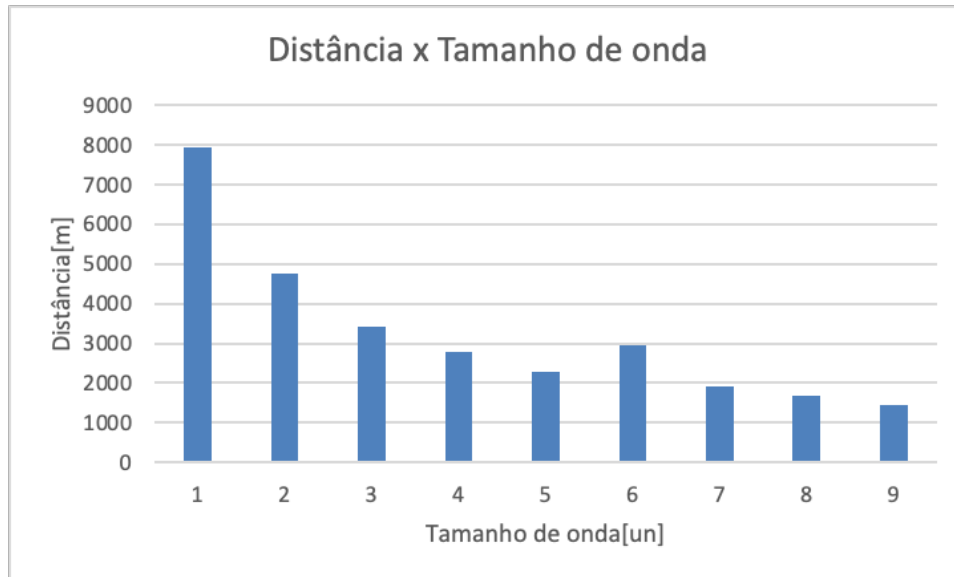
Fonte: Autor (2021)

Os resultados indicam uma redução de 38,6% na distância ao aplicar a estratégia de wave picking enquanto a estratégia de clusterização espacial a redução foi de 54,1%, ao se comparar com a distância obtida sem aplicar estratégias de movimentação.

Ou seja, os impactos na redução da distância total ainda são significativos, todavia, é possível perceber que com a demanda maior, a redução na distância não apresenta o mesmo impacto do que com a demanda reduzida.

Agora que o ajuste de limiar foi identificado como 17 metros, um segundo teste foi realizado para verificar o impacto de alterar o tamanho das ondas com esse limiar fixado. O resultado obtido foi o seguinte:

Figura 30 - Ajuste de onda com melhor limiar de distância do cenário 1



Fonte: Autor (2021)

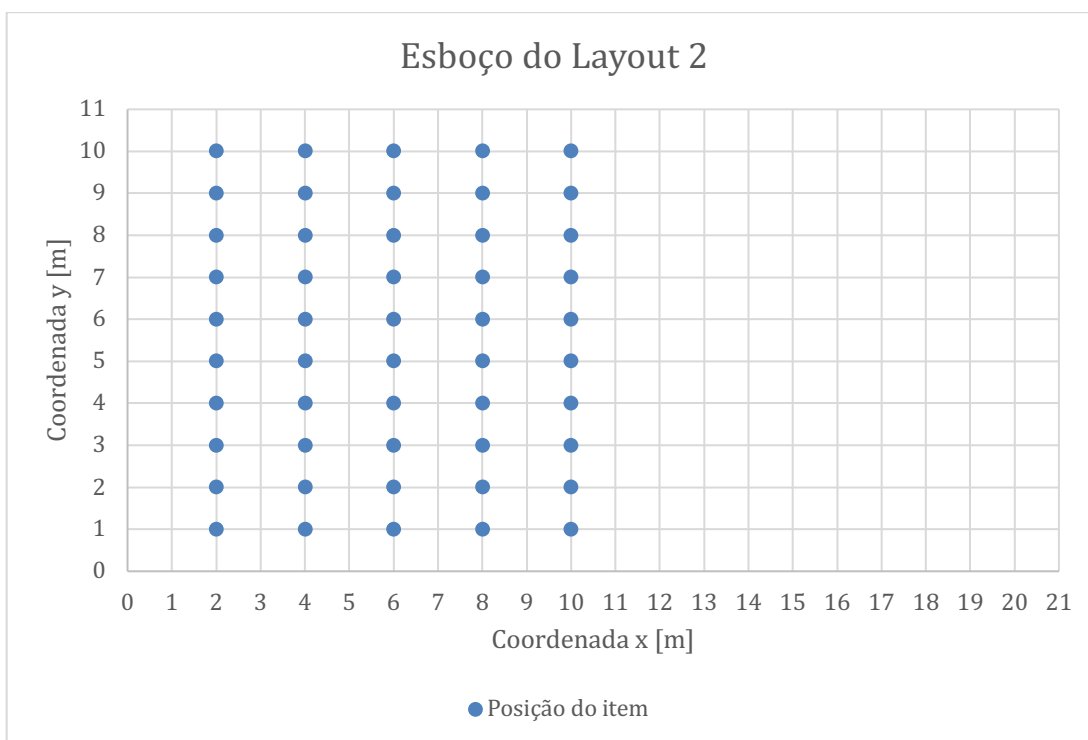
Constatou-se que com o limiar da distância fixado em 17 metros e o tamanho de onda em 9 pedidos por onda de coleta a distância total foi alterada de 1666 metros para 1454 metros.

5.2.1 Geração dos itens e suas localizações no cenário 2

A estrutura física do layout do armazém escolhido para o segundo cenário de simulação consiste na reestruturação do layout anterior para metade da área, o número de corredores reduzido a metade da quantidade anterior. Sendo assim, a quantidade de corredores agora é de cinco, seu comprimento se mantém o mesmo, porém considera-se que os itens serão alocados em um mesmo ponto de coleta de um outro item, ou seja, cada ponto de coleta representará agora as coordenadas de dois itens ao invés de apenas um.

Para melhor visualizar o layout definido para a segunda simulação a Figura 31 abaixo ilustra um esboço do armazém considerado para as análises, utilizando a mesma escala do esboço mostrado no primeiro cenário.

Figura 31- Esboço do Layout 2



Fonte: Autor (2021)

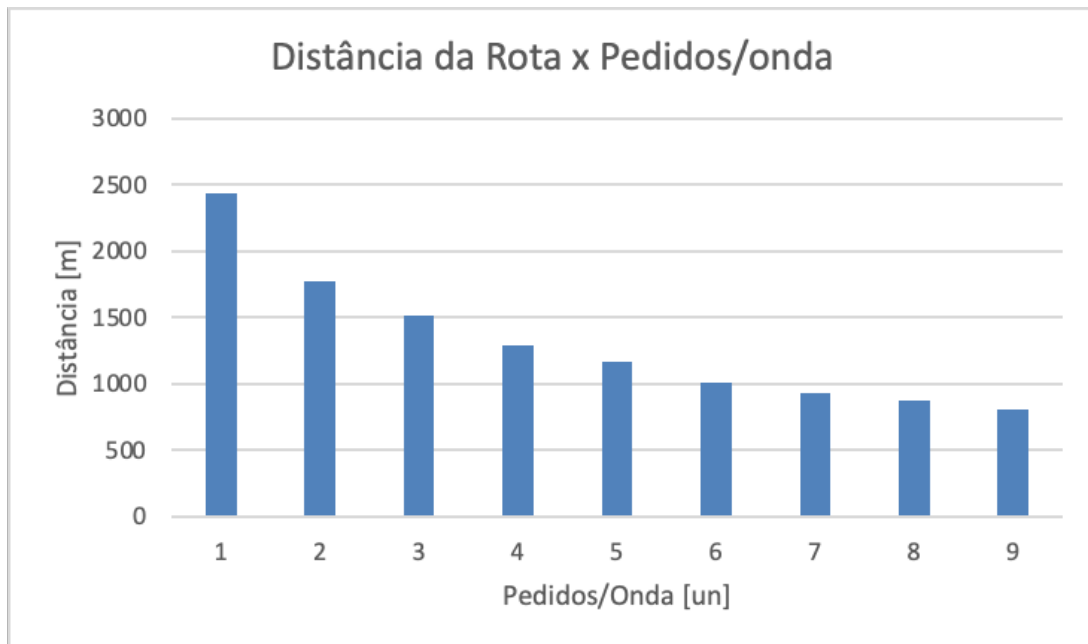
5.2.2 Geração da lista de pedidos do cenário 2

A lista de pedidos utilizada para a segunda simulação é a mesma que a considerada para a primeira simulação, no qual aproximadamente 80% dos pedidos são direcionados a 20% dos itens do armazém e a mesma regra é aplicada para as quantidades de itens por pedidos, limitado entre uma e duas unidades, conforme explicado no tópico 5.1.2.

5.2.3 Primeira análise do cenário 2

Da mesma maneira que a menor distância foi dada por ondas de nove pedidos por coleta no primeiro cenário, o segundo cenário também gerou a mesma resposta, conforme mostrado na Figura 32 abaixo.

Figura 32 - Resultado em colunas da primeira análise do Layout 2



Fonte: Adaptado de Saci (2020)

A distância total obtida para a melhor resposta de nove itens por onda é de 800 metros e pode significar uma redução de até 67,2% ao se comparar com o pior cenário de *picking* de um item por rota, com a distância total percorrida de 2440 metros.

O resultado detalhado das rotas obtidas a partir da separação dos pedidos em grupos de nove é demonstrado na Tabela 7 abaixo.

Tabela 7 - Resultado detalhado da primeira análise do Layout 2

N° da Onda	Distância da Rota	Caminho Percorrido	Pedidos/Onda
0	68	[[0, 0], (2, 1), (2, 4), (2, 6), (2, 8), (4, 10), (4, 5), (6, 9), (6, 4), (10, 9), [0, 0]]	9
1	68	[[0, 0], (4, 4), (4, 5), (6, 9), (2, 8), (10, 7), (8, 2), (8, 4), [0, 0]]	9
2	56	[[0, 0], (2, 1), (2, 3), (2, 6), (6, 2), (8, 3), (10, 7), [0, 0]]	9
3	60	[[0, 0], (2, 3), (2, 6), (4, 4), (8, 3), (8, 6), (8, 9), (10, 1), [0, 0]]	9
4	62	[[0, 0], (2, 1), (2, 5), (2, 6), (6, 4), (8, 2), (10, 6), (10, 8), (10, 9), [0, 0]]	9
5	52	[[0, 0], (2, 3), (2, 6), (6, 9), (6, 2), (10, 1), (4, 4), [0, 0]]	9
6	50	[[0, 0], (2, 1), (2, 4), (2, 5), (2, 6), (4, 1), (6, 4), (10, 1), (10, 4), [0, 0]]	9
7	62	[[0, 0], (2, 1), (2, 5), (2, 7), (6, 9), (8, 9), (8, 6), (10, 4), (4, 4), [0, 0]]	9
8	74	[[0, 0], (2, 3), (2, 5), (4, 1), (4, 4), (4, 10), (6, 6), (6, 2), (8, 3), (10, 9), [0, 0]]	9
9	52	[[0, 0], (2, 1), (2, 3), (2, 4), (2, 8), (8, 9), (8, 3), (8, 2), (10, 1), (6, 4), [0, 0]]	9
10	46	[[0, 0], (6, 2), (8, 3), (8, 9), (10, 9), (10, 6), (10, 1), [0, 0]]	9
11	54	[[0, 0], (2, 3), (2, 4), (2, 6), (4, 8), (6, 1), (6, 2), (6, 4), (8, 4), [0, 0]]	9
12	54	[[0, 0], (2, 1), (4, 4), (6, 4), (6, 9), (8, 9), (8, 4), (8, 3), (10, 1), [0, 0]]	9
13	42	[[0, 0], (2, 6), (10, 9), [0, 0]]	9

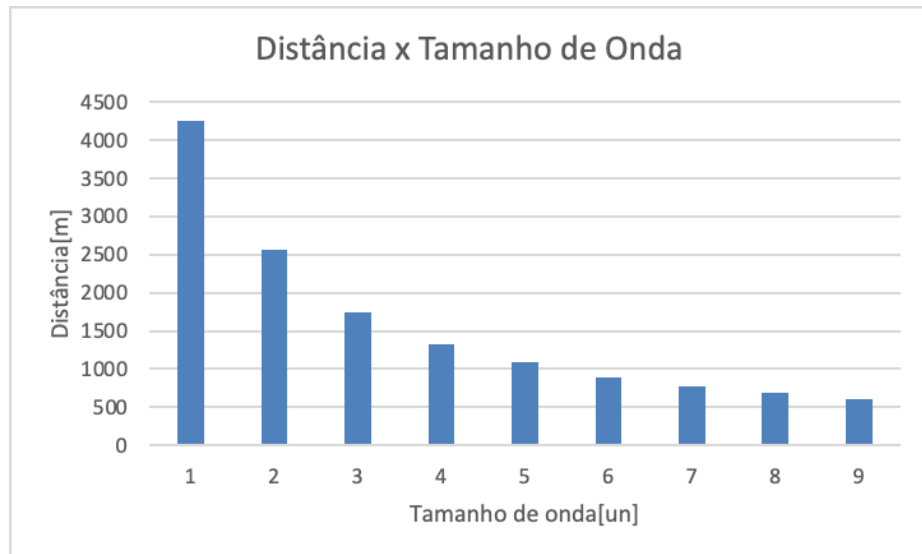
Fonte: Autor (2021)

Da mesma maneira que ocorreu na primeira análise, alguns caminhos não indicaram nove pontos de coleta, além do ponto de expedição de ida e volta. Isso acontece devido a presença de itens repetidos na mesma onda de coleta e, para a última onda, o valor também é reduzido pois a onda é preenchida com os itens restantes e, novamente, não apresentou valor próximo a nove pontos de coleta com apenas dois pontos além do ponto de expedição.

5.2.3.1 Primeira análise do cenário 2 com demanda alterada

Foi considerado também um tamanho de lista de pedidos duplicado e, o resultado obtido foi como mostrado abaixo.

Figura 33 - Resultado da primeira análise do Layout 2 com demanda alterada



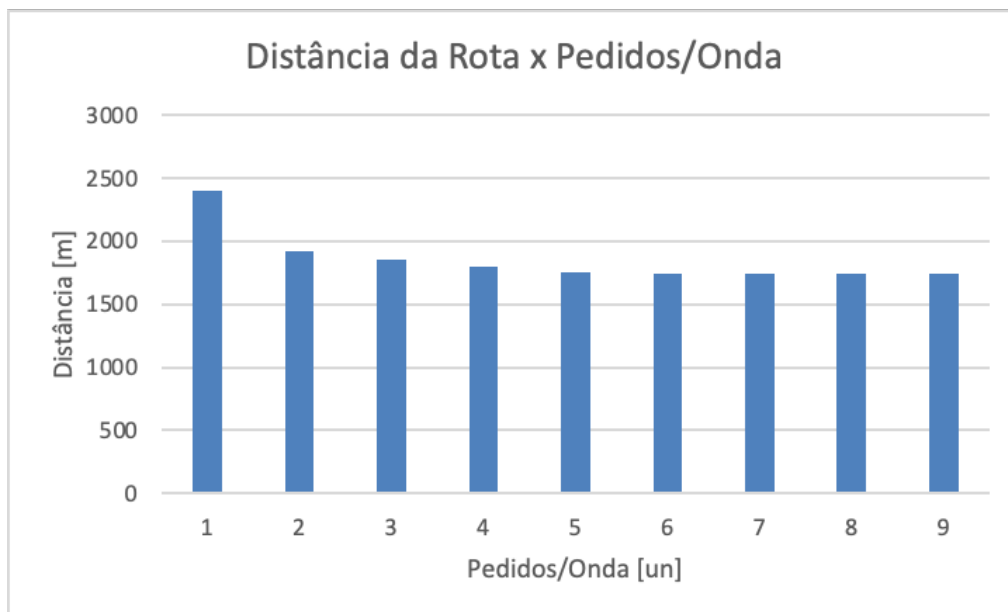
Fonte: Adaptado de Saci (2020)

Novamente, a melhor configuração é de nove pedidos por onda, como nos cenários anteriores.

5.2.4 Segunda análise do cenário 2

A aplicação da estratégia de *clustering* espacial para determinação da separação dos pedidos, utilizando-se a mesma lista de pedidos da análise 1, foi iniciada com limiar de distância inicial de dez metros a fim de se obter uma resposta para o tamanho de onda adequado para continuar a simulação testando-se diferentes limiares de distância observando-se a distância total obtida. O resultado dessa primeira etapa é demonstrado na Figura 34 abaixo.

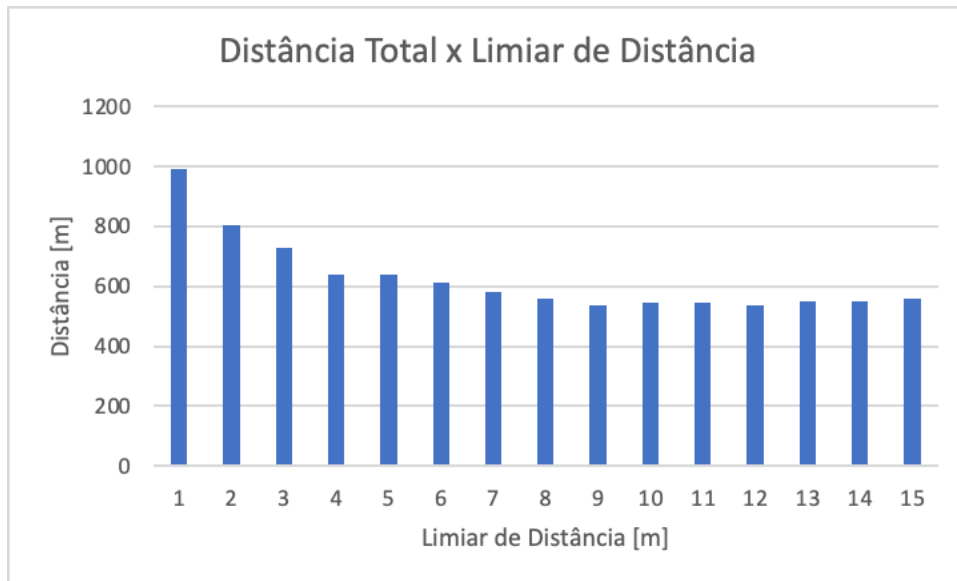
Figura 34 - Resultado em colunas da segunda análise do Layout 2



Fonte: Adaptado de Saci (2020)

A partir de seis pedidos por onda é possível observar que o incremento no tamanho da onda já não afeta a distância da rota. Dessa forma, para esse segundo cenário de simulação o tamanho de onda considerado é alterado de nove itens por onda para seis. Portanto, a configuração inicial para testar os valores de limiar de distância e as distâncias de rota obtidas é de seis pedidos por onda e, os resultados obtidos são demonstrados abaixo na Figura 35.

Figura 35 - Ajuste de limiar de distância da segunda análise do Layout 2



Fonte: Adaptado de Saci (2020)

Como o layout considerado para o segundo cenário de simulação apresenta dimensões consideravelmente reduzidas, a faixa de valores determinados para teste também foram reduzidas sendo agora de 1m a 15m, a fim de não tratar o layout como apenas um *cluster*. O resultado obtido mostra que os limiares de distância de 9 e 12 metros resultam nos melhores valores de distância total, de 538 metros.

A fim de se analisar o potencial de redução na distância total a partir da técnica aplicada para esse segundo cenário, a Tabela 8 abaixo demonstra as respostas obtidas no caso extremo com limiar de distância de apenas um metro e a resposta obtida.

Tabela 8 - Dados do ajuste de limiar de distância do Layout 2

Limiar da distância[m]	Distância total percorrida[m]	N° de Clusters[un]	Tamanho de Onda[un]
1	990	16	6
9	538	7	6

Fonte: Autor (2021)

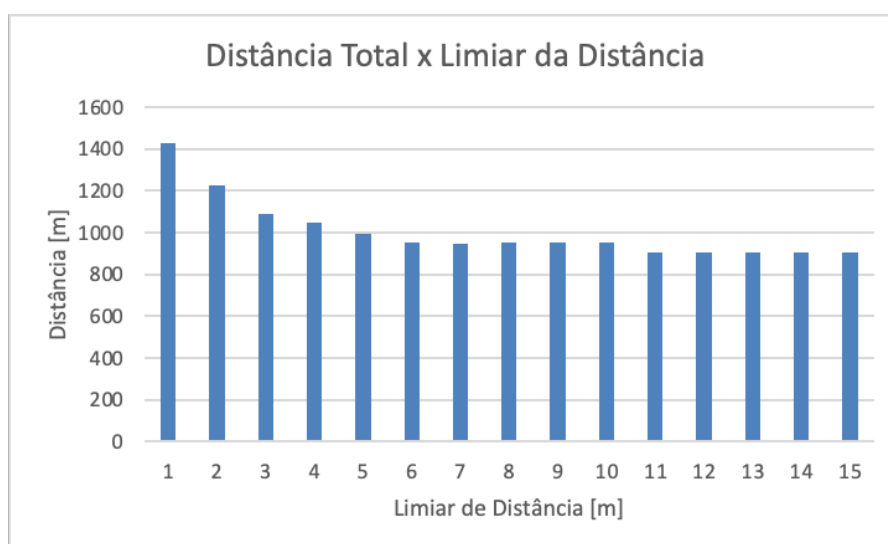
Novamente, foi verificado que com o menor limiar, a quantidade de clusters é maior do que quando o limiar da distância é consideravelmente maior. O resultado obtido mostrou que é

possível reduzir aproximadamente 45,6% da distância total a partir do ajuste eficiente do limiar da distância para criação dos *clusters*.

5.2.4.1 Segunda análise do cenário 2 com demanda alterada

Para uma simulação considerando-se o dobro da lista de pedidos, foi verificado o seguinte resultado:

Figura 36 - Ajuste de limiar de distância da segunda análise do Layout 2 com demanda alterada

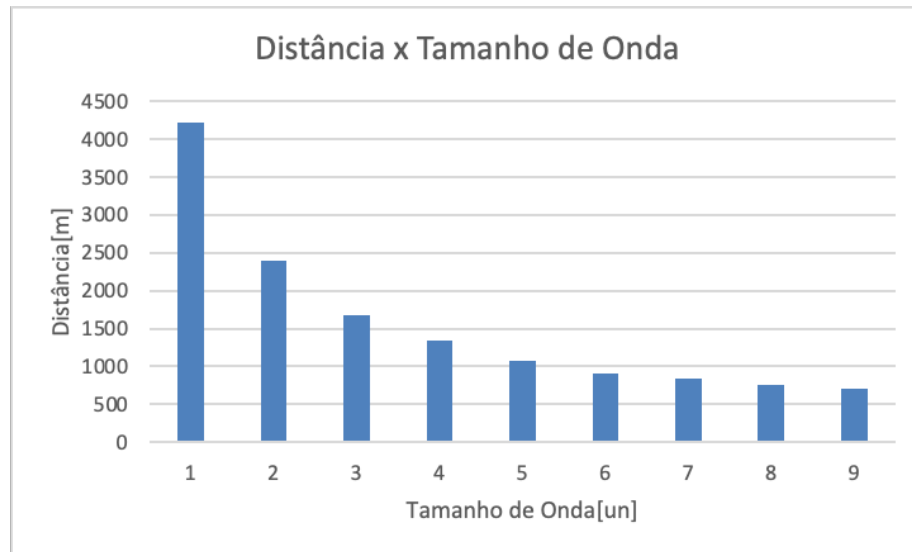


Fonte: Adaptado de Saci (2020)

Nesse caso, o ajuste do limiar da distância resulta na melhor resposta quando está configurado a 11 metros, resultando em uma distância total de 904 metros.

Com o limiar de distância configurado para sua melhor resposta de 11 metros, um segundo teste foi realizado para verificar o impacto de alterar o tamanho das ondas com esse limiar fixado, o resultado obtido é mostrado na Figura 37 abaixo.

Figura 37 - Ajuste de onda com melhor limiar de distância do cenário 2



Fonte: Adaptado de Saci (2020)

De maneira similar, ao aumentar o tamanho de onda para nove pedidos por *batch*, foi possível otimizar a resposta passando de 904 metros quando o tamanho é igual a seis, para 702 metros.

5.2.5 Comparação das Análises do Cenário 2

A simulação para o segundo cenário também implicou em impactos significativos na distância total das rotas do operador.

Da mesma maneira demonstrada no primeiro cenário, as mesmas três situações são consideradas para a análise:

- Sem aplicação de estratégias;
- Melhor resposta obtida a partir da aplicação da estratégia de *wave picking*;
- Melhor resposta obtida a partir da aplicação da estratégia de clusterização espacial.

Figura 38 - Resultados das estratégias no Layout 2

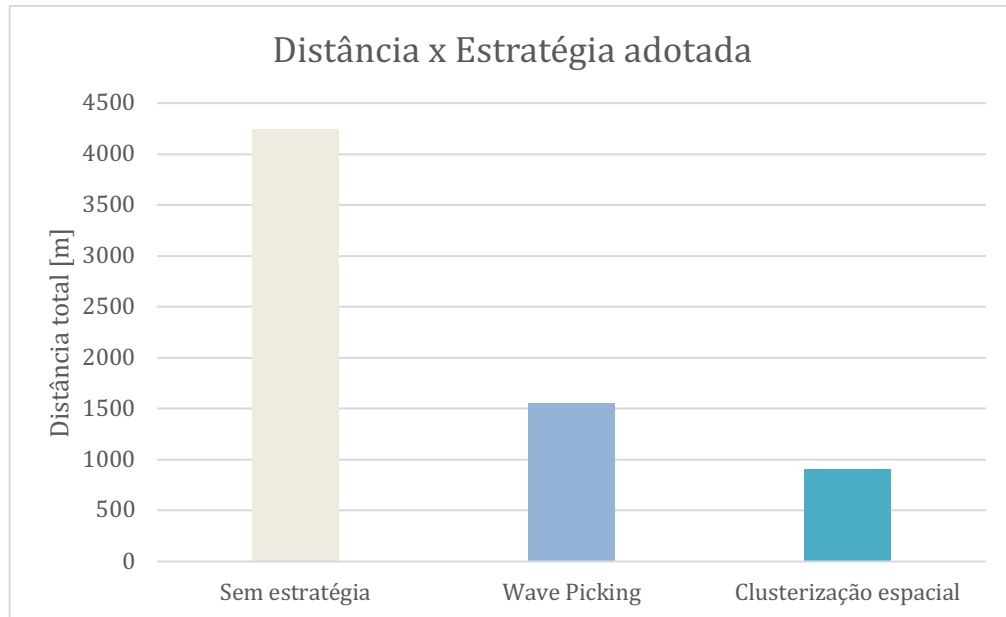


Fonte: Autor (2021)

Os resultados indicam uma redução de 67,2% na distância com a aplicação da estratégia de *wave picking* e 78% com a estratégia de clusterização espacial, ao se comparar com a distância obtida sem aplicar estratégias de movimentação.

Os resultados obtidos no cenário 2 ao alterar a demanda para o dobro do tamanho, apresentou diferenças pouco significativas ao se comparar com os resultados da demanda anterior, com uma redução de 63,2% na estratégia de *wave picking* e 78,7% na clusterização espacial, como mostrado na Figura 39 abaixo.

Figura 39 - Resultados das estratégias no Layout 1 com demanda alterada



Fonte: Autor (2021)

5.3 AVALIAÇÃO

Os incrementos nas variáveis de ajustes nos dois cenários resultaram em respostas com impactos similares, sendo possível identificar soluções que reduziram significativamente a distância com valores acima de 77% ao se comparar a melhor solução com a movimentação sem a aplicação de estratégias como guia das coletas.

As similaridades nos resultados têm relação com a estrutura física dos armazéns, os espaços de movimentação, exposição dos itens e ponto de expedição considerados. Mas, ao se comparar os resultados dos cenários entre eles, levando em conta que ambos armazenam a mesma quantidade de itens e receberam a mesma lista de pedidos, a distância total varia do cenário 1 de 812 metros para 538 metros no cenário 2, o que é bastante impactante.

5.3.1 Avaliação da estratégia de *wave picking*

A Tabela 9 abaixo resume os resultados obtidos com a aplicação da estratégia de *wave picking* nos dois cenários.

Tabela 9 - Resultados da estratégia de wave picking

Cenário	Distância [m]	Tamanho de Onda[un]
1	1226	9
2	800	9

Fonte: Autor (2021)

A primeira análise mostra que a mesma estratégia aplicada no segundo layout resulta em aproximadamente 65,2% do caminho que seria percorrido no primeiro layout.

5.3.2 Avaliação da estratégia de clusterização espacial

A estratégia se mostrou extremamente eficaz em ambos os casos. As etapas de configuração inicial para determinação do tamanho de onda e de testes com incremento gradual do limiar de distância, foram fundamentais para encontrar a melhor configuração em cada caso. A Tabela 10 abaixo resume os resultados obtidos na aplicação da clusterização espacial em seu melhor ajuste em cada cenário.

Tabela 10 - Resultados da estratégia de clusterização espacial

Cenário	Distância [m]	Tamanho de Onda[um]
1	812	9
2	538	6

Fonte: Autor (2021)

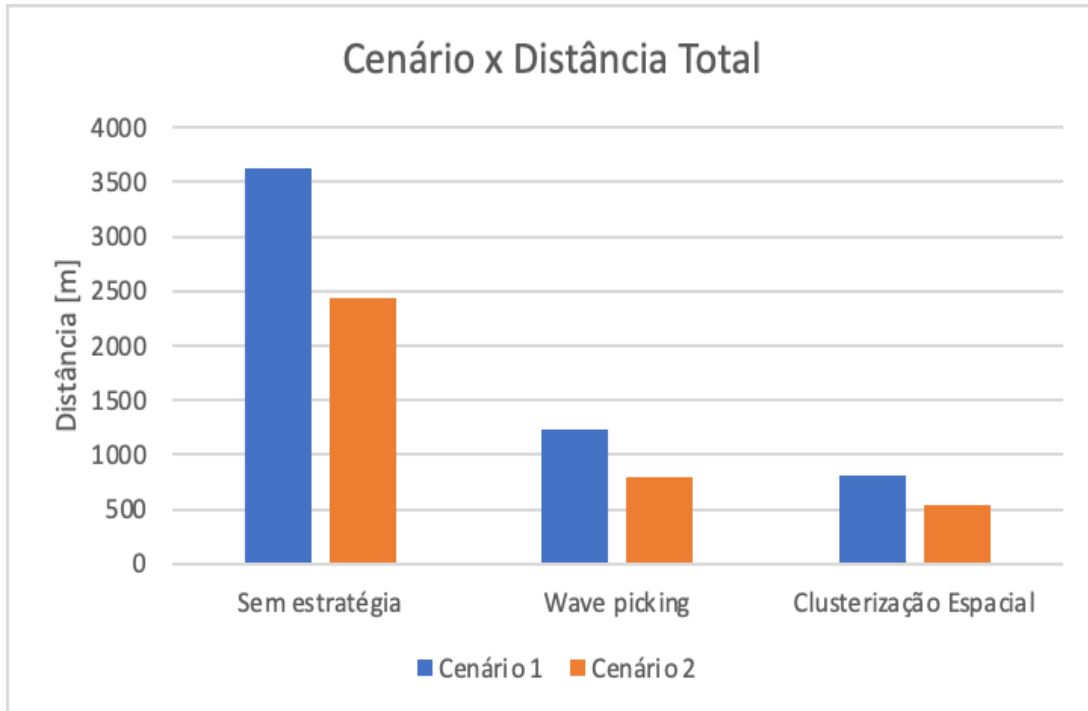
Essa segunda estratégia aplicada no segundo layout resultou em aproximadamente 66,2% do caminho que seria percorrido no primeiro layout.

5.3.3 Avaliação Global

Intuitivamente, o segundo cenário resultaria em uma distância menor por apresentar dimensões reduzidas e a possibilidade de coletar dois itens distintos em um mesmo ponto, ou

seja, o operador percorreria caminhos menores que no layout anterior em metade dos itens e, o resultado das simulações comprovou isso, conforme ilustrado na Figura 40 em seguida.

Figura 40 - Resultado global da aplicação das estratégias



Fonte: Autor (2021)

O resultado geral indica um comportamento similar após a simulação das estratégias nos dois cenários, com quedas acentuadas nas distâncias. E, o layout menor apresentou distâncias totais menores. Porém, não foi considerado os esforços extras necessários ao se optar por um layout com dimensões reduzidas e mesmas quantidades de itens, no qual a reposição de estoque é mais frequente e exige tempo e recurso a mais, o que significa mais custo e, em uma avaliação geral, esse aspecto não deve ser ignorado. Além disso, ao optar por reduzir o layout é possível reduzir também a necessidade de reposição de itens de acordo com a forma como os itens estão distribuídos fisicamente no local (paletes, meio pallet, prateleiras) a partir da recorrência associada ao item, o que poderia impactar em redução de custos, mas não está dentro do escopo do presente trabalho.

Por fim, a aplicação da técnica de separação de pedidos em ondas em conjunto com a clusterização espacial se mostrou eficaz no sentido de guiar a movimentação de um operário responsável pela atividade de *picking* como ferramenta para o curto prazo. Outras ações como

mudança no layout e posicionamento estratégico dos itens de acordo com sua recorrência podem impactar significativamente na redução da distância percorrida no *picking*, porém, a demanda pode variar drasticamente em períodos curtos e, essas alterações exigem mais tempo e recursos para suas implementações e podem se tornar inviáveis.

6 CONCLUSÃO

A atividade de *picking* é uma das atividades mais críticas e com restrição de tempo dentre as operações de um armazém e, pode afetar significativamente toda a cadeia de suprimentos. O trabalho realizado teve como objetivo propor alternativas de movimentação de um operador do armazém que minimize a distância total de viagem no processo de separação de pedidos, o que por sua vez minimiza os custos operacionais globais. Isso foi alcançado por meio de otimização e simulação, considerando-se o problema de roteirização para um único operador (“Single Picker Routing Problem-SPRP”) em um modelo bidimensional de armazém com layouts e disposição dos itens diferentes, no qual a partir de uma lista de pedidos e locais de armazenamento com suas respectivas coordenadas, as técnicas de wave picking e clusterização espacial foram aplicadas para a separação de pedidos e determinação da rota de coleta, a fim de se encontrar soluções com a mínima distância total de viagem na atividade de picking para então guiar a movimentação do operador de acordo com o layout do armazém e o tamanho da lista de pedidos recebida.

As alternativas de movimentação de um operador foram simuladas e dois cenários foram considerados, o primeiro para um armazém de 10 corredores de 10 metros de comprimento, e o segundo para um armazém de 5 corredores com 10 metros de comprimento, ambos com a mesma demanda estabelecida pela lista de pedidos e variação da demanda. A simulação da estratégia de wave picking com o objetivo de ajustar o tamanho ideal de onda de coleta que retorne a mínima distância de viagem mostrou-se eficaz, proporcionando reduções na distância de até mais da metade do que seria sem separação por ondas de coleta nos dois cenários.

As respostas tiveram impactos mais satisfatórios ao realizar a combinação do tamanho de onda ideal da primeira estratégia com a segunda estratégia de clusterização espacial, o que indicou reduções acima de 70% na distância total de viagem.

Sem o ajuste de tamanho de onda e limiar de distância para criação de zonas de coleta, as técnicas não apresentariam o mesmo impacto. Outras ações como mudança no layout e posicionamento estratégico dos itens de acordo com sua recorrência são alternativas atraentes, mas em períodos curtos podem ser inviáveis.

Por fim, apesar dos resultados positivos encontrados, o trabalho foi realizado em formato de simulação, algumas etapas foram simplificadas e os dados foram integrados de

maneira simples. Na prática, existem inúmeros desafios e etapas a serem cumpridas antes da implementação das técnicas simuladas nesse trabalho, muitos armazéns podem não apresentar um sistema de gerenciamento de armazém (WMS) que são essenciais para a coleta de dados e processamento das estratégias demonstradas, além disso, mesmo quando existe um sistema instalado, a etapa de tratamento e integração com outras estruturas também podem representar um grande desafio.

Para um futuro trabalho recomenda-se o estudo de como otimizar o layout do armazém a partir da redução dos espaços utilizados para aumentar a produtividade na atividade de *picking*. Outra sugestão de trabalho direcionado a continuação dos desafios encontrados no presente trabalho, é a simulação de alternativas de movimentação para mais de um operador atuando simultaneamente e com quantidade máxima de itens por pedido aumentada.

REFERÊNCIAS

EDWARDS, J. B.; MCKINNON, A. C.; CULLINANE, S. L. **Comparative analysis of the carbon footprints of conventional and online retailing: A “last mile” perspective.** *International Journal of Physical Distribution & Logistics Management*, v. 40, n. 1/2, p. 103–123, 2 fev. 2010.

LING-FENG, H.; LIHUI, T. **The optimum design of a warehouse system on order picking efficiency.** *International Journal of Advanced Manufacturing Technology*, v.28, n.5/6, p.626-637, 2006.

POON, T.; CHOY, K.; CHOW, H., LAU, H.; CHAN, F.; HO, K. **A RFID case-based logistics resource management system for managing order-picking operations in warehouses.** *Expert Systems with Applications*, v.36, n.4, p.8277–8301, 2009.

GOMES, C.; RIBEIRO, P. **Gestão da cadeia de suprimentos integrada à tecnologia da informação.** São Paulo: Pioneira Thomson Learning, 2004.

KIM, C.; YANG, K; KIM, J. **A strategy for third-party logistics systems: a case analysis using the blue ocean strategy.** *Omega*, v. 36, n.4, p.522–534, 2008.

NOVAES, A. **Logística e gerenciamento da cadeia de distribuição: estratégia, operação e avaliação.** Rio de Janeiro: Elsevier, 2004.

CHOUERI, A. **Uma proposta de heurística para o single picker routing problem considerando restrições de empilhamento máximo,** 2018. UFPR.

R. d. Koster, T. Le-Duc, and K. J. Roodbergen, **“Design and control of warehouse order picking: A literature review,”** *European Journal of Operational Research*, vol. 182, no. 2, pp. 481–501, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0377221706006473>

(NOUD) GADEMANN, A. J. R. M.; VAN DEN BERG, J. P.; VAN DER HOFF, H. H. **An order batching algorithm for wave picking in a parallel-aisle warehouse.** *IIE Transactions*, v. 33, n. 5, p. 385–398, maio 2001.

ALDSTADT, J. **Spatial Clustering.** In: FISCHER, M. M.; GETIS, A. (Eds.). . *Handbook of Applied Spatial Analysis.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 279–300.

HENN, S. **A New Mathematical Programming Formulation for the Single-Picker Routing Problem in a Single-Block Layout.** p. 32, [s.d.].

ZUNIC, E. et al. **A generic approach for order picking optimization process in different warehouse layouts.** 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). Anais... In: 2018 41ST INTERNATIONAL CONVENTION ON INFORMATION AND COMMUNICATION TECHNOLOGY, ELECTRONICS AND MICROELECTRONICS (MIPRO). Opatija: IEEE, maio 2018. Disponível em: <<https://ieeexplore.ieee.org/document/8400183/>>. Acesso em: 17 set. 2021.

Ebit - WebShoppers - O mais completo relatório sobre o e-commerce. Disponível em: <<https://company.ebit.com.br/webshoppers/webshoppersfree>>. Acesso em: 12 set. 2021.

KARASEK, J. **An Overview of Warehouse Optimization.** International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems, v. 2, n. 3, 22 nov. 2013.

SACI, S. **Improve Warehouse Productivity using Order Batching with Python.** Disponível em: <<https://towardsdatascience.com/optimizing-warehouse-operations-with-python-part-1-83d02d001845>>. Acesso em: 12 set. 2021.

SACI, S. **Improve Warehouse Productivity using Spatial Clustering with Python Scipy.** Disponível em: < <https://towardsdatascience.com/optimizing-warehouse-operations-with-python-part-2-clustering-with-scipy-for-waves-creation-9b7c7dd49a84> >. Acesso em: 12 set. 2021.

SHETTY, N.; SAH, B.; CHUNG, S. H. **Route optimization for warehouse order picking operations via vehicle routing and simulation.** SN Applied Sciences, v. 2, n. 2, p. 311, fev. 2020.

HENN, S.; KOCH, S.; WÄSCHER, G. **Order Batching in Order Picking Warehouses: A Survey of Solution Approaches.** In: MANZINI, R. (Ed.). . Warehousing in the Global Supply Chain. London: Springer London, 2012. p. 105–137.

HOFFMAN, K. L.; PADBERG, M. **Traveling Salesman Problem.** In: GASS, S. I.; HARRIS, C. M. (Eds.). . Encyclopedia of Operations Research and Management Science. New York, NY: Springer US, 2001. p. 849–853.

PEERLINCK, K.; GOVAERT, T.; LANDEGHEM, H. V. **A DESIGN METHOD FOR PARTS PICKING ZONES IN A MANUFACTURING ENVIRONMENT.** p. 4, [s.d.].

DE KOSTER, M. B. M. **Warehouse Assessment in a Single Tour.** In: MANZINI, R. (Ed.). . Warehousing in the Global Supply Chain. London: Springer London, 2012. p. 457–473.

Karim, N.H., Abdul Rahman, N.S.F., Md Hanafiah, R., Abdul Hamid, S., Ismail, A., Abd Kader, A.S. and Muda, M.S. (2021), "**Revising the warehouse productivity measurement indicators: ratio-based benchmark**", Maritime Business Review, Vol. 6 No. 1, pp. 49-71. <https://doi.org/10.1108/MABR-03-2020-0018>

KOLINSKI, A.; SLIWczynski, B. EVALUATION PROBLEM AND ASSESSMENT METHOD OF WAREHOUSE PROCESS EFFICIENCY. p. 14, 2015.

APÊNDICE A – Repositório com os códigos das análises do layout 1 e 2

<https://github.com/erikspanhol/apendiceTCC>