

Optimization of Management Zones Shape Files

Letícia Leite Caetano, Jocival Dantas Dias Junior, André Ricardo Backes, Hericles Felipe Ferraz and Mauricio Cunha Escarpinati
School of Computer Science
Federal University of Uberlândia
Uberlândia, Brazil
leticialeitecaetano17@gmail.com and mauricio@ufu.br

Abstract—The growing use of technologies in favor of Precision Agriculture enables the application of different strategies in a crop and seeks to increase production, reduce costs and reduce damage to the environment. To keep up with the need to increase productivity and still reduce costs with farming as much as possible, the approach of applying inputs in a targeted manner based on the classification of regions is increasingly used, as are the results obtained in [9]. In optimizing these results, some points were identified that could be improved in relation to the vector data of the generated Management Zones, such as overlapping between different zones, invalid geometries, and a very large amount of points, which add unnecessary complexity to the file. This work proposes an algorithm that aims to optimize these Management Zone results in a *shapefile*, and aims to correct invalid geometries, reduce the number of points that define the shapes of the zones, and the correction of overlapping regions so that zones with lesser vigor have priority. In addition, an adjustment of the spacing between the geometries is made while correcting the overlap between different zones. As a result, a new *shapefile* is created, composed only of valid geometries, fewer points, and no overlaps between different Management Zones. Specialists evaluated the results obtained and indicated them as adequate to solve the problem.

Index Terms—Computational Geometry. Management Zones. Overlay . Points Simplification. Invalid Geometries.

I. INTRODUÇÃO

Obter dados precisos, significativos que levem a maior produtividade e eficiência nas lavouras é um dos grandes motivadores do emprego da tecnologia no campo, pois, com o crescimento constante da população e a redução dos recursos disponíveis, há uma necessidade de se otimizar o uso desses recursos, para que seja possível atender a demanda crescente por alimentos, [10]. Dessa forma, é necessário que os métodos de produção se adaptem e se tornem mais e mais precisos de modo que os recursos disponíveis sejam aproveitados no seu potencial máximo, [2].

Nesse contexto, a Agricultura de Precisão (AP), tem como propósito identificar variações na lavoura, através do uso de técnicas geoespaciais e sensores, para que assim, essas variações possam ser tratadas de forma localizada e orientada aos problemas detectados [13]. Além disso, possibilita a identificação da variabilidade da área analisada, permitindo diferentes formas de manejo, para atender cada região utilizando apenas os recursos necessários [7]. Assim, problemas que prejudicam qualitativa e quantitativamente a produção, são reduzidos, evitando gastos desnecessários e estimando com maior precisão o rendimento final.

O Índice de Vegetação (IV) pode ser obtido através de imagens de uma lavoura e tem como função principal investigar os atributos espectrais geralmente relacionados a características biofísicas da mesma, assim, podem ser considerados como indicadores de vigor e de crescimento das plantas, [8]. A partir de valores de IV calculados, pode se ter uma representação exibida em grupos, com características semelhantes, gerando um tipo de indicador de campo baseado em agrupamentos espaciais chamado de Zonas de Manejo (ZM). As ZM caracterizam regiões que variam do menor valor de vigor ao maior e indicam o estado vegetativo da plantação, [9].

Computacionalmente, tais regiões de manejo são registradas como forma geométricas (*splines* ou *shapefile*) e estas, em muitos casos apresentam problemas de concorrência, sobreposição, entre outros. A ocorrência de regiões, nas ZM, contendo geometrias inválidas, impede a utilização de algumas funcionalidades oferecidas por bibliotecas de manipulação de dados vetoriais, e o mesmo ocorre em Sistemas de Informações Geográficas. Além disso, a ocorrência de sobreposições entre diferentes zonas, dificulta o levantamento correto de dados, principalmente com relação ao valor da área das regiões em um talhão. Outro problema, está relacionado a grande quantidade de pontos utilizados para definir as formas dessas zonas, sendo que elas podem ser representadas de forma semelhante, utilizando uma menor quantidade de pontos.

O *ESRI shapefile* ou *shapefile (.shp)*, é um formato simples e não topológico, desenvolvido e regulamentado pela ESRI, que é um empresa que trabalha com sistemas de informação geográfica. Esse formato engloba arquivos contendo dados geoespaciais vetorizados. Tais arquivos armazenam informações referentes a posição, o formato e os atributos dos dados geográficos das formas, assim como explicado em [5].

Para resolver os problemas propostos, nesse trabalho foram utilizadas técnicas e ferramentas que atuam com a manipulação de arquivos vetoriais. E para facilitar a manipulação desses dados foi utilizada a *Geospatial Data Abstraction Library (GDAL)* ou *GDAL/OGR* como é mais conhecida. Essa biblioteca mescla soluções que trabalham com dados *raster*, que anteriormente atuava de forma separada como *GDAL* e dados vetoriais, anteriormente conhecido apenas como *OGR*, [12]. Para que não fique confuso, neste trabalho usaremos apenas *OGR* para se referir a parte dessa biblioteca que trabalha com dados vetorias.

A correção dos problemas listados tem grande importância para que se possa trabalhar com os dados vetoriais das ZM de maneira mais otimizada e estimar valores mais precisos com relação à delimitação de cada zona, evitando áreas repetidas e facilitando a manipulação e uso desses arquivos. Assim, neste trabalho é proposta uma metodologia que visa analisar e otimizar a estrutura do *shapefile* das ZM produzidas a partir do trabalho desenvolvido por Rezende et. al. [9].

II. TRABALHOS CORRELATOS

Rezende et al [9] apresentam em seu trabalho uma metodologia de monitoramento da saúde da vegetação através da utilização de técnicas de agrupamento de dados, a partir dos resultados de IV, que fragmenta a cultura em uma quantidade de áreas preestabelecida, considerando as semelhanças de cada região.

O algoritmo em [3], realiza a simplificação de linhas, alcançando bons resultados e preservando bem a simetria das formas.

Em [6] os autores propõem algo semelhante ao realizar operações sobre polígonos através de um algoritmo que realiza o cálculo da área sobreposta entre dois polígonos irregulares. Para isso, os polígonos são decompostos respectivamente no menor número de polígonos convexos, após isso, cada par de polígonos convexos sobrepostos de duas partições resultantes são recortados e sua área de sobreposição é calculada.

Já em [1], os autores propõem uma solução alternativa ao método de varredura do plano, e procura resolver problemas relacionados a sobreposição de polígonos de forma mais eficiente, abordando uma grande variedade de formatos de polígonos, entre eles, polígonos degenerados, sobrepostos, auto-sobrepostos, disjuntos ou com furos.

Para esse trabalho, procura-se uma forma de otimizar a estrutura do *shapefile* das Zonas de Manejo criadas a partir do trabalho apresentado em [9]. Assim, o primeiro problema atacado foi número abundante de pontos que definem a estrutura das ZM, o que já foi tratado em [3], entretanto, como exposto em [11], pode produzir picos que não existem na forma original e generalizações desequilibradas dependendo da tolerância aplicada. Sabendo disso, o método proposto visa corrigir as bordas das geometrias, para evitar que o resultado final apresente picos ou deformações que não existam na geometria original.

Para o problema de sobreposição entre diferentes feições do *shapefile*, que é reforçado após a redução da quantidade de pontos, este trabalho realiza a eliminação das regiões sobrepostas, diferentemente do que é feito em [6], que realiza o cálculo da área de sobreposição e não especificamente a remoção dessas regiões.

Em comparação com o trabalho de [1], o trabalho proposto também trata áreas sobrepostas a partir de diversos formatos de polígonos e ainda trata a redução da quantidade de pontos.

III. MATERIAL E MÉTODOS

Neste capítulo será apresentada a estrutura e o funcionamento dos métodos propostos, os quais tem por propósito a

otimização dos vetores de ZM obtidos a partir do trabalho apresentado por [9]. Para isso, serão executadas as seguintes etapas: correção de geometrias inválidas, redução da quantidade de pontos e por fim a correção de sobreposições e ajustes de lacunas.

Para a execução das etapas descritas foi proposto o desenvolvimento de métodos que visam tratar cada um desses casos, como pode-se observar na Figura 1. Nessa figura, cada uma dessas etapas são apresentadas como blocos e esses blocos são descritos de A até E, sendo A o bloco que engloba o método que verifica a validade das geometrias, B é responsável pela correção das geometrias inválidas, C verifica se os pontos já foram simplificados, D é responsável por reduzir os pontos e, por fim, o Bloco E corrige a sobreposição entre as feições e ajusta as lacunas.

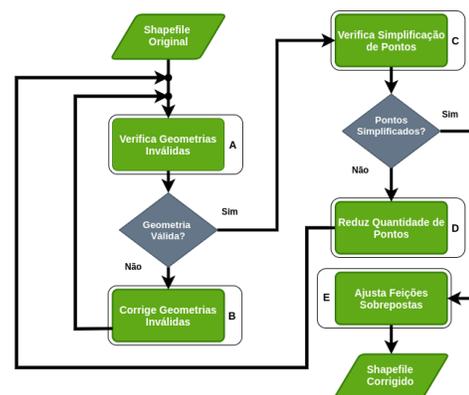


Figura 1. Fluxograma representando as etapas do algoritmo. Imagem: fonte própria.

A. Correção de Geometrias Inválidas

Com o *shapefile* aberto, o foco é na identificação e tratamento dos problemas relacionados a geometrias inválidas, e para isso, cada feição no arquivo é percorrida dentre os dados vetoriais da camada, e para cada uma dessas feições, identifica-se a referência da geometria correspondente. Assim é feita uma verificação de validade, como apresentado no Bloco A da Figura 1, para todas as geometrias presentes na camada analisada através da função *verifica_geometria_valida*, que verifica se existe problemas relacionados com auto-interseção, auto-contato, duplicação de vértice ou polígono com menos de três vértices e caso alguns desses problemas for identificado, assim como no Bloco B da Figura 1, a função *corrige_geometria* é chamada passando como parâmetro a geometria inválida identificada e retorna uma geometria sem problemas de validade. Após a execução desse processo para cada feição, os problemas que tornam as geometrias inválidas são corrigidos.

B. Redução da Quantidade de Pontos

Para a redução da quantidade de pontos que definem as ZM, foi necessário a definição de um conjunto de valores para diferentes tolerâncias de distância. Cada tolerância representa o limite de distância utilizado como base no processo de simplificação.

Assim, foi realizada uma análise empírica para a escolha dos melhores valores relacionados a tolerância de distância a serem testados. Para isso, foi realizada uma experimentação, onde observou-se os extremos relacionados a tolerância de distância, e assim, um intervalo foi definido sem que as formas ficassem muito deformadas ou que as mesmas possuíssem um nível exagerado de detalhes, esses limites foram escolhidos por especialistas, os quais definiram quais extremos eram mais convenientes para a resolução do problema, considerando a máxima redução de pontos e a mínima redução de pontos. Devido ao sistema de coordenadas utilizado os valores de tolerância de distância são utilizados em graus. Dessa forma, os valores escolhidos para análise foram 0.0000010° , 0.0000009° , 0.0000008° , 0.0000007° , 0.0000006° , 0.0000005° , 0.0000004° , 0.0000003° , 0.0000002° , e 0.0000001° . Quanto mais alto o valor passado como limite de distância, menos pontos restarão na camada final e quanto mais baixo for esse valor, mais pontos restarão no arquivo de saída.

Sabendo disso, como os pontos que definem as ZM ainda não foram simplificados, o próximo passo consiste na simplificação desses pontos, Bloco D da Figura 1, e para a execução dessa etapa, para cada zona, obtém-se a geometria correspondente e para todas as formas presentes em cada geometria de cada zona é aplicada a função responsável pela de simplificação, *simplifica_qtd_pontos*. Através dessa função é possível diminuir a quantidade de vértices dada uma determinada tolerância de distância, sendo um processo semelhante ao algoritmo *Douglas-Peucker*, [4].

Como resultado da operação de simplificação, novas formas são geradas, as quais possuem formato semelhante às formas originais, entretanto, a quantidade de vértices utilizada para representar a geometria, é reduzida, de acordo com cada uma das tolerâncias investigadas.

Após a redução da quantidade de pontos podem ocorrer problemas de auto-interseção, ou algum outro que cause erro de validade nas geometrias de cada zona, e esses problemas podem ser agravados dependendo do valor tolerado para a distância na simplificação dos vértices. Quanto maior esse valor, menos pontos são mantidos e quanto menor for o valor de tolerância, mais pontos permanecerão na definição da estrutura da forma. Para garantir que esses problemas não persistam para as próximas etapas, caso haja alguma geometria inválida, a operação que faz a correção das mesmas é chamada, Bloco B da Figura 1, e esses problemas são corrigidos antes de avançar para o próximo passo.

C. Correção de Sobreposições e Ajustes de Lacunas

Nessa etapa, após conferir que o processo de simplificação foi realizado com sucesso, Bloco C da Figura 1, cada geometria de cada zona é comparada com todas as outras geometrias de cada feição, fazendo isso com o intuito de verificar a ocorrência de interseções entre essas geometrias.

Na verificação das interseções, a geometria analisada é comparada com a geometria resultante da aplicação de um *buffer* sobre a geometria a ser testada. Os valores aplicados

para identificar a melhor distância, passada como parâmetro para a função de *buffer*, foram 0.0000005 e 0.0000008 , que foram obtidos através de experimentos e considerados aceitáveis pelos especialistas. e para o parâmetro de quantidade de seguimentos, o valor 2 foi utilizado, pensando em obter a menor quantidade de pontos definindo as curvas. Esse processo é realizado com o objetivo de se obter regiões maiores de interseção, para posteriormente realizar a correção dos espaços entre os polígonos que, como dito anteriormente, podem ficar desfigurados após a redução da quantidade de pontos. Mas as análises de interseções, entre as geometrias originais, ainda são realizadas para se obter os dados reais relacionados a sobreposição entre diferentes feições.

Caso uma interseção com uma geometria, resultante da aplicação de um *buffer*, seja identificada, é realizada uma operação de diferença, retirando essa geometria obtida como resultado da aplicação do *buffer*, da região sobreposta da geometria, em que a feição correspondente tenha o maior IV e mantendo a região da zona que apresenta menor IV, sendo esta última privilegiada no processo de eliminação de sobreposição, Bloco E da Figura 1. Após isso um novo *shapefile* é gerado e o resultado para cada zona é salvo. Esse processo ocorre até que todas as zonas sejam comparadas e corrigidas.

No processo de correção das sobreposições, regiões serão comparadas e a área de interseção entre elas será mantida apenas para a feição que possuir o menor IV. Além disso, nesse processo de correção de sobreposição já é feito o ajuste dos espaços entre as formas da camada, para se obter um resultado mais agradável.

IV. RESULTADOS E DISCUSSÕES

Nesse capítulo, são apresentados os resultados obtidos durante a investigação dos problemas propostos para esse trabalho. Para tanto, foram realizados inúmeros testes com diferentes parâmetros para avaliar o algoritmo de redução da quantidade de pontos em um conjunto de dados vetoriais, a fim de definir qual o parâmetro limite para que não houvesse perda significativa da qualidade da forma. Além disso, também será mostrado os dados relacionados as correções de sobreposições encontradas entre diferentes zonas, correção de erros que tornam as geometrias inválidas, e como resultados adicionais os espaçamentos encontrados entre diferentes formas serão ajustados para evitar locais com lacunas desfiguradas.

A. Dados Analisados

Para a realização dos testes foram utilizados 10 arquivos do tipo *shapefile*, os quais armazenam dados vetoriais referentes as ZM de um talhão de 1,45 hectares, obtido como resultado com base no trabalho apresentado por Rezende et al [9]. Cada arquivo tem armazenado um conjunto de dados com diferentes configurações relacionadas a quantidade de zonas, variando de 2 a 10 regiões. Sendo assim, os resultados foram obtidos utilizando os métodos propostos a partir desses 10 arquivos *shapefile* de entrada, cada um contendo respectivamente 2, 3, 4, 5, 6, 7, 8, 9 e 10 zonas.

B. Correção de Geometrias Inválidas

Com a aplicação desse método, nos casos em que o arquivo original continha problemas que tornavam a geometria inválida, o resultado obtido para os testes realizados, foi uma camada livre desses erros para cada uma das zonas verificadas, como pode ser observado na Tabela I. Para obter esses resultados, foi feita uma investigação dos dados vetoriais, através de uma função, com o propósito de verificar se as geometrias de cada zona eram válidas, antes e depois da aplicação do método, investigando a quantidade de zonas que possuíam geometrias inválidas e, além disso, também foram levantados dados referentes a quantidade de pontos antes e depois da execução dessa etapa. Para os resultados obtidos nessa investigação foi criada a Tabela I, que representa a quantidade de zonas e a quantidade de pontos antes e depois da aplicação desse método. Na primeira parte da tabela, pode-se observar que todas as zonas, de cada arquivo, que possuíam algum erro de validade foram corrigidas e na segunda parte, observa-se que houve uma leve redução na quantidade de pontos, após a realização dessa correção.

TABLE I
QUANTIDADE DE ZONAS APRESENTANDO ERROS DE GEOMETRIA INVÁLIDA E QUANTIDADE DE PONTOS ANTES E DEPOIS DA PRIMEIRA APLICAÇÃO DO MÉTODO DE CORREÇÃO DE AUTO-INTERSEÇÕES, CONSIDERANDO ARQUIVOS DE 2 A 10 ZONAS

Nº. Zonas	Zonas com geometrias inválidas		Quantidade de pontos	
	Antes	Depois	Antes	Depois
2	2	0	68088	67315
3	3	0	118026	116747
4	4	0	153285	151688
5	5	0	200768	198706
6	6	0	216140	213900
7	7	0	260417	257365
8	8	0	307752	304378
9	9	0	355899	351637
10	10	0	393875	388804

C. Redução da Quantidade de Pontos

Como resultado para a aplicação dos métodos propostos, temos a Figura 2(a) que demonstra graficamente os resultados obtidos, relacionados a quantidade de pontos. Esse gráfico apresenta a comparação do número de vértices presentes no arquivo original em cada configuração de zona, com a quantidade de pontos nas camadas de saída, obtidas como resultado dessa etapa, para cada quantidade de zonas e para cada tolerância aplicada. Isso foi feito, considerando a redução da quantidade de pontos em cada um dos 10 arquivos testados, e dez diferentes tolerâncias de distância, onde cada uma foi passada como parâmetro para função de simplificação, e aplicada para cada um dos arquivos que possuem de 2 a 10 zonas.

A partir desses dados foi possível verificar o percentual de pontos, que foi reduzido após a aplicação da simplificação, levando em consideração cada tolerância de distância, aplicada

para cada arquivo de 2 a 10 zonas, assim como mostrado na Figura 2(b).

Outro importante resultado, está associado diretamente à simplificação dos pontos, que é a redução do tamanho do arquivo *.shp*, do tipo *shapefile*, que contém as características das geometrias, sendo o maior arquivo do conjunto de arquivos gerados. Para a realização dessa análise foi observado os arquivos gerados após a simplificação dos pontos, para cada configuração de zona e para cada limite de distância. Para isso, os resultados dessa etapa foram salvos, apenas para comparação, e cada arquivo gerado foi analisado e teve seu tamanho extraído. Nesse arquivo foi observado que conforme diferentes tolerâncias são aplicadas e a quantidade de pontos é reduzida, o tamanho desse arquivo também é diminuído, como mostrado na Figura 2(c).

Na Figura 2(d), pode-se verificar graficamente o percentual de redução do tamanho do arquivo após a aplicação da redução de pontos. Observa-se também maiores reduções de pontos conforme a tolerância de distância aumenta e menores reduções conforme esse valor fica mais baixo.

Através das Figuras 2(b) e 2(d), nota-se que mesmo utilizando a menor tolerância de distância comparada, a qual preserva uma maior quantidade de pontos, ainda assim o percentual de redução do tamanho do arquivo e o percentual de redução da quantidade de pontos, indicam uma redução de mais de 40% para os dois casos. Além disso, entre os valores testados para a tolerância de distância, o maior valor de tolerância consegue reduzir mais de 90% da quantidade de pontos e mais de 90% do tamanho do arquivo.

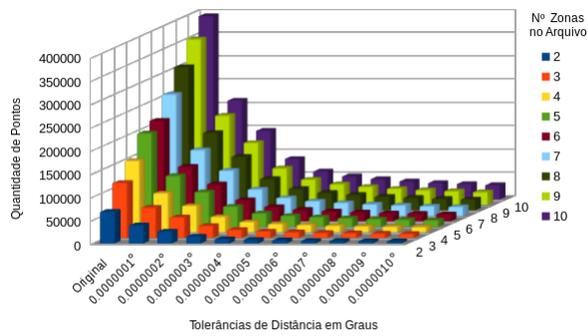
D. Correção de Geometrias Inválidas após Simplificação de Pontos

Após a realização da redução da quantidade de pontos, alguns problemas relacionados a validade das geometrias podem ser encontrados, mesmo já tendo passado pela etapa de correção de geometrias inválidas anteriormente. Assim é necessário que sejam verificados novamente, já que a simplificação de pontos pode gerar novos erros. Então, nessa etapa, a camada analisada é verificada novamente em busca de possíveis erros.

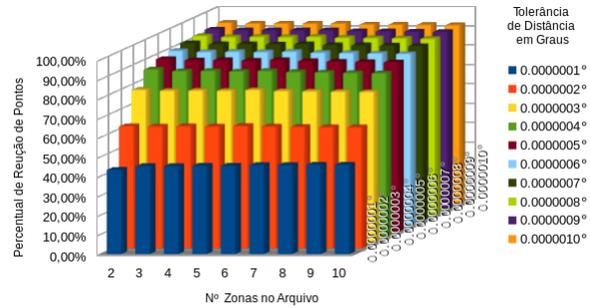
Após essa verificação, os resultados obtidos indicam que o processo de simplificação gerou geometrias inválidas, as quais foram identificadas em apenas em seis dos arquivos gerados, ocorrendo em apenas uma das feições, para os arquivos com 6, 7 e 10 zonas e tolerâncias aplicadas de 0.0000006° e 0.0000005° . Para todos os casos de erro, a correção de geometrias inválidas foi aplicada novamente e como resultado final, nenhuma zona apresentou esse tipo de erro.

E. Correção das Sobreposições e ajuste de espaços entre as formas

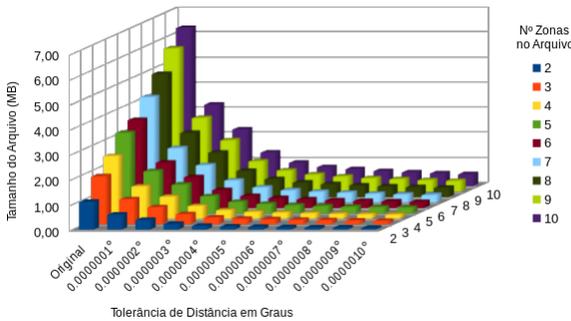
Conforme observado nos resultados apresentados nas seções anteriores, a metodologia proposta resolveu os problemas relacionados com sobreposições de diferentes feições e, ao mesmo tempo, a sobreposição de bordas foi ajustada. No entanto, os parâmetros empregados no processo de redução dos vértices,



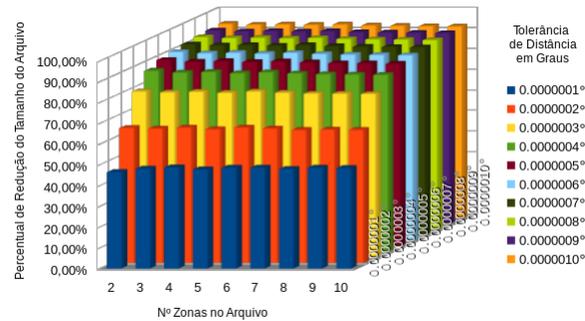
(a) Demonstração da redução da quantidade de pontos após aplicação do processo de simplificação.



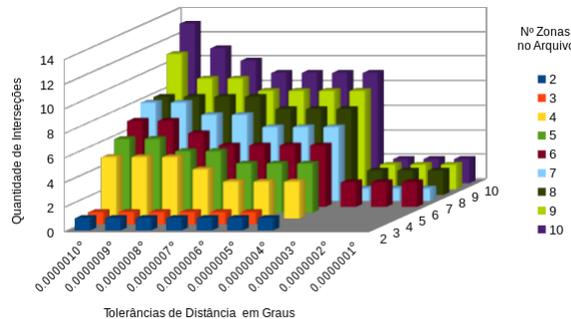
(b) Demonstração do percentual de pontos reduzidos considerando arquivos de 2 a 10 zonas e tolerâncias de simplificação no intervalo de 0.0000001° a 0.0000001° . Imagem: fonte própria.



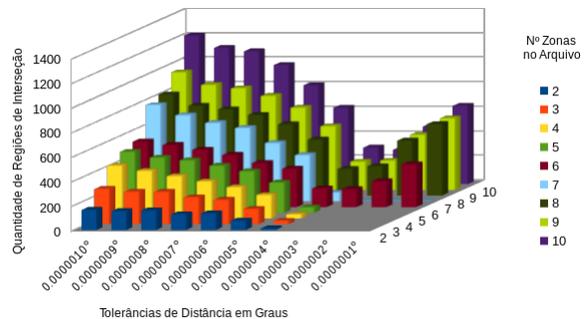
(c) Demonstração do tamanho dos arquivos *shapefile*, que contém dados vetoriais, considerando cada arquivo de 2 a 10 zonas e tolerâncias de simplificação no intervalo de 0.0000001° a 0.0000001° . Imagem: fonte própria.



(d) Demonstração do percentual de tamanho de arquivo reduzido considerando arquivos de 2 a 10 zonas e tolerâncias de simplificação no intervalo de 0.0000001° a 0.0000001° . Imagem: fonte própria.



(e) Demonstração da quantidade de interseção nas geometrias de cada zona, considerando cada arquivo de 2 a 10 zonas e tolerâncias de simplificação no intervalo de 0.0000001° a 0.0000001° .



(f) Demonstração da quantidade de regiões de interseção nas geometrias de cada zona, considerando cada arquivo de 2 a 10 zonas e tolerâncias de simplificação no intervalo de 0.0000001° a 0.0000001° .

Figura 2. Apresentação dos resultados obtidos

bem como no algoritmo de diferença de buffer utilizado no tratamento da sobreposição de borda, geram deformações nos resultados finais obtidos, ou seja, na forma final do shape que representa a zona de manejo. Desta forma, fez-se necessário aferir, sob a ótica de especialistas, quais os valores ideais para tais parâmetros de forma que os problemas de sobreposição e a quantidade excessiva de vértices fossem resolvidos e ainda assim a qualidade final do *shape* se mantivesse sob um aspecto desejável.

Para a realização de tal investigação foram apresentados a 4

especialistas 9 shapes representando diferentes ZM. Para cada um desses shapes foi apresentado aos especialistas sua forma original e as formas resultantes do processo de redução de vértices com cada um dos parâmetros de tolerância testados como mostra a Figura 2(a). Além disso, foram apresentados dois diferentes valores para distância de *buffer*.

Após colher a opinião dos especialistas, conforme apresentado na tabela II, para dois dos 4 especialistas os valores 0.0000008° como distância de *buffer* e tolerância de distância para simplificação da forma apresentaram os mel-

hores resultados, para um deles a melhor configuração seria 0.0000005° para *buffer* e tolerância de distância e para o último 0.0000005° para *buffer* e 0.0000010° para tolerância de distância. Com isso, tem-se um indicador de que os valores com maior aceitação são os mais adequados para a resolução do problema.

TABLE II
REPRESENTAÇÃO DOS MELHORES RESULTADOS INDICADOS PELOS ESPECIALISTAS.

Distância de <i>buffer</i>	Tolerância da Distância
0.0000008°	0.0000008°
0.0000008°	0.0000008°
0.0000005°	0.0000005°
0.0000005°	0.0000010°

Na Figura 2(e), é apresentado graficamente as ocorrências de interseções, encontradas para cada arquivo de 2 a 10 zonas e levando em consideração cada tolerância aplicada anteriormente na etapa de simplificação da quantidade de pontos, ou seja, essa tabela mostra a quantidade de interseções encontradas ao se comparar uma feição com todas as outras feições presentes no arquivo. Além disso, foi possível a identificação de regiões de interseção que ocorrem para cada feição e a contagem das mesmas, como é apresentado de forma gráfica na Figura. 2(f).

Após a aplicação do método de correção de sobreposições, foi feita uma nova verificação para cada *shapefile*, que foram gerados como resultados de todas as etapas realizadas para a otimização das ZM. Nessa verificação, para cada tolerância de distância investigada, não foi encontrada nenhuma ocorrência de sobreposição em nenhum dos casos analisados.

V. CONCLUSÃO

Como resultados das etapas executadas pelo algoritmo, obteve-se um arquivo *shapefile* contendo uma camada de dados vetoriais livre de problemas relacionados a geometrias inválidas, com uma quantidade de pontos reduzida e sem sobreposição entre diferentes feições. Para a definição dos melhores valores de tolerância de simplificação e de distância de *buffer*, foram consultados 4 especialistas, onde dois deles, indicaram os valores 0.0000008° como distância de *buffer* e o mesmo valor para a tolerância de distância na simplificação. Para um dos especialistas, a melhor configuração seria 0.0000005° para *buffer* e tolerância de distância e de acordo com o último especialista consultado, 0.0000005° para *buffer* e 0.0000010° para tolerância de distância. Mesmo que não tenham sido realizadas muitas análises, é possível identificar um indício de que os valores com maior aceitação possam ser os mais indicados para o problema.

Um dos problemas que não foi tratado e que merece atenções futuras está relacionado aos espaços existentes entre as formas das zonas, os quais não impactam tanto para esse trabalho mas podem ser um problema quando se pensa na utilização desses arquivos por sistemas que controlem a aplicação de insumos de forma direcionada.

REFERENCES

- [1] Samuel Audet et al. “Robust and efficient polygon overlay on parallel stream processors”. In: Nov. 2013, pp. 304–313. DOI: 10.1145/2525314.2525352.
- [2] I. F. Carvalho. *Agricultura de precisão: Boletim Técnico*. Tech. rep. Brasília: Mapa/ACS, 2011.
- [3] DAVID H. DOUGLAS and THOMAS K. PEUCKER. “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature”. In: *Cartographica: The International Journal for Geographic Information and Geovisualization* 10 (Dec. 1973). DOI: 10.3138/FM57-6770-U75U-7727. URL: <https://doi.org/10.3138/FM57-6770-U75U-7727>.
- [4] *Douglas-Peucker-Algorithmus*. <https://de.wikipedia.org/wiki/Douglas-Peucker-Algorithmus>. Acessado em 16/06/2019. 2018.
- [5] ESRI ESRI. “Shapefile technical description”. In: *An ESRI white paper* (1998).
- [6] Zi-qiang Li, Yan He, and Zhuo-jun Tian. “Overlapping Area Computation between Irregular Polygons for Its Evolutionary Layout Based on Convex Decomposition”. In: *JSW* 7 (2012), pp. 485–492.
- [7] L. Z PES et al. *Projeto Aquarius/Cotrijal – Pólo de Agricultura de Precisão na Região Alto Jacuí – RS*. Tech. rep. 2º Congresso Brasileiro de Agricultura de Agricultura de precisãoESALQ/USP, 2006.
- [8] FJ Ponzoni. “Comportamento espectral da vegetação”. In: *Sensoriamento remoto: reflectância de alvos naturais* 1 (2001), pp. 157–199.
- [9] G. Rezende Silva et al. “Definition of Management Zones Through Image Processing for Precision Agriculture”. In: *2017 Workshop of Computer Vision (WVC)*. Oct. 2017, pp. 150–154. DOI: 10.1109/WVC.2017.00033.
- [10] Paolo Tropicchio et al. “Towards smart farming and sustainable agriculture with drones”. In: *2015 International Conference on Intelligent Environments*. IEEE. 2015, pp. 140–143.
- [11] Mahes Visvalingam and Simon Herbert. “A Computer Science Perspective on the Bendsimplification Algorithm”. In: *Cartography and Geographic Information Science* 26.4 (1999), pp. 253–270. DOI: 10.1559/152304099782294177. eprint: <https://doi.org/10.1559/152304099782294177>. URL: <https://doi.org/10.1559/152304099782294177>.
- [12] Erik Westra. *Python geospatial development*. Packt Publishing Birmingham, 2010, pp. 48–53.
- [13] Chunhua Zhang and John Kovacs. “The application of small unmanned aerial systems for precision agriculture: A review”. In: 13 (Dec. 2012).