JOMTA
Journal of Mathematics: Theory and Applications
Vol. 6, No. 1, 2024, pp. 75 - 96
P-ISSN: 2685-9653 e-ISSN: 2722-2705
DOI: 10.31605/jomta.v6i1.3511

Valorant Haven Strategy Using BIP and Weighted Graph

Ermelinda Benna Kireyna[1*], Jovian Dian Pratama[2], Mauliddino Rizky Pratama[3], Sri Lutfiya Dwiyeni[4], Apni Diyanti[5], Bernardinus Rico Dewanto[6], Sunarsih[7]

[1,2,3,4,5,6,7] *Department of Mathematics, Faculty of Science and Mathematics, Diponegoro University, Semarang 50275, Indonesia*

Corresponding email: kireynalinda@gmail.com*

Abstract

Strategy in a competitive video game is needed to reach a successful game, for example, Valorant. Route planning is one of the strategies in games. In consideration of making a new strategy, this research develops a binary integer programming (BIP) model to generate an optimal route depending on passable paths, travel time, kills, and survivability. By using POM QM for Windows to compute the model, we obtained optimal modified routes that can be combined in the role and agent composition.

Keywords: *Valorant, Haven, binary integer programming, post-match strategy, route planning*

Received :25-12-2023, Revised :22-03-2024, Accepted :31-03-2024

1. Introduction

In recent years, video games have been growing significantly. One of the most popular video games in terms of most played games as of May 2023 is Valorant [1]. Valorant is a massively multiplayer online first-person shooter with a 5v5 character-based game. This game uses the anti-terrorist versus terrorist concept where the anti-terrorist (defender) will restrain the terrorist from entering the bomb site, and the terrorist (attacker) will make its way to plant the bomb on the site. Valorant also restricts players' behavior, such as cheating, communication abuse, queue dodging, idle or away from the keyboard (AFK), and team sabotage. If a player keeps violating the rules, there will be a warning, temporary ban, or permanent ban, depending on their behavior [2]. A match in Valorant is divided into two halves: the first half is where the total score of both sides reaches 12, then the teams will swap sides, and the second half is where the teams fight for the remaining rounds. A team that scores 13 winning rounds first will win the match, or there will be overtime rounds if the teams have been through 24 rounds where the score of both teams is tied to 12-12. For overtime, a team can only win the match if their score is two scores higher than the other team, meaning that overtime is endless. However, overtime can be stopped if both teams agree to draw the match. [3] – [5].

In a match, each round has 30 to 45 seconds of buy phase where the players discuss their strategy and 100 seconds for the round to be played. With this information, the defenders can defend the site or eliminate the attackers to win the round. If the attackers planted the bomb, the round will have 40 seconds of extra time. During that period, the defenders will try to defuse the bomb, and the attackers will try to defend it until it explodes [6], [7]. In order to dominate the match, the players must be familiar with the map. There are ten maps that can be played in Valorant, which are Ascent, Bind, Haven, Split, Icebox, Breeze, Pearl, Lotus, Fracture, and Sunset. Haven and Lotus are special maps with three bomb sites [8]. Before entering the match, players must choose agents that suit their strategy on a particular map. Agents in Valorant have four roles: Duelist, Controller, Initiator, and Sentinel.

Duelist opens the path by seeking out engagements with the enemies, impacting the team by focusing on clearing paths and sites and eliminating most of the enemies. Controller creates room for the team by blocking the enemy's vision through the smoke. Initiator provides information with their abilities, and most of them have flashes to help Duelist enter the site. Sentinel mainly focuses on defense with traps and barricades (Heath, 2023; Valorant Wiki, n.d.). Thus, a strategy to have excellent map control and team coordination is necessary.

A strategy can be made through route planning, which determines the players' route for a successful round. Methods such as integer programming have been widely used in route planning. Some researchers solved routing problems with single or multi-criteria [11] – [22], generated evacuation routes [23] – [25], and found the shortest paths with the help of Dijkstra's Algorithm [26]. Research about a rotation route in Valorant was made for defenders in Breeze. However, the route was made using only Dijkstra's algorithm with time as one of its variables [27].

This paper uses the Valorant Champions Tour 2022: Istanbul (VCT 2022) as our study case. Based on the tournament, 11 of 16 teams started as attackers, and most of them lost the half-match in Haven. Teams that lost once as attackers are EDG, TL, FUR, FNC, BOOM, PRX, LEV, and OPTIC. A team that lost twice as attackers is DRX. Considering Haven is a special map with three sites (shown in Figure 1) and most of the teams lost the first half match as attackers, we propose a strategy using a binary integer programming (BIP) model and weighted graph that copes with the strategy limitations which are the number of passable paths, obtained kills, survivability, and travel time. Inspired by the previous research about route planning, we choose the BIP model and weighted graph to generate a strategy of route planning that considers variables that affect the strategy and not simply route for the shortest paths. We also expect to produce one or more proposed routes for attackers by maximizing the BIP model and forming weighted graphs from the data. Since we must collect data from previous matches, this strategy will be used as a post-match strategy. It aims to help the teams have a better strategy for the following competitions.

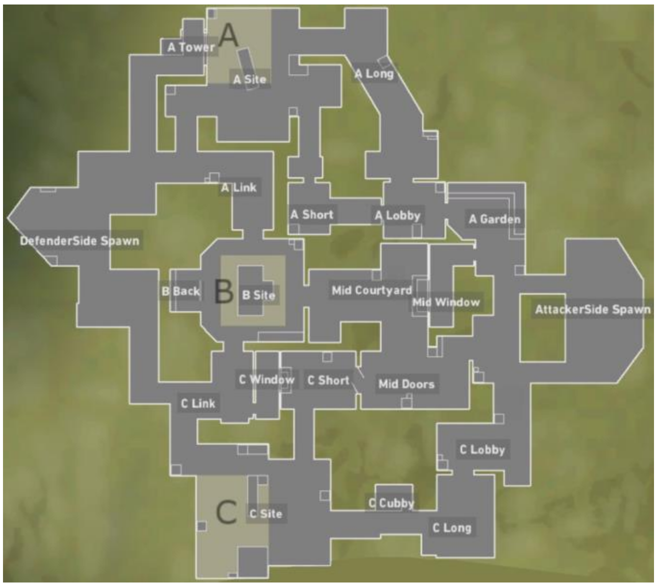


Figure 1. Map of Haven

The paper is structured as follows. Section 2 describes the method and BIP model. The results of the research are in Section 3. Lastly, the conclusion is in Section 4.

2. Methods

The research method contains how agents and roles will be selected, graph construction, how the data will be used in the model, and a definition of BIP.

2.1 Agents and Roles Selection

We surveyed Haven's preferred role composition among the players and classification of role composition in teams.

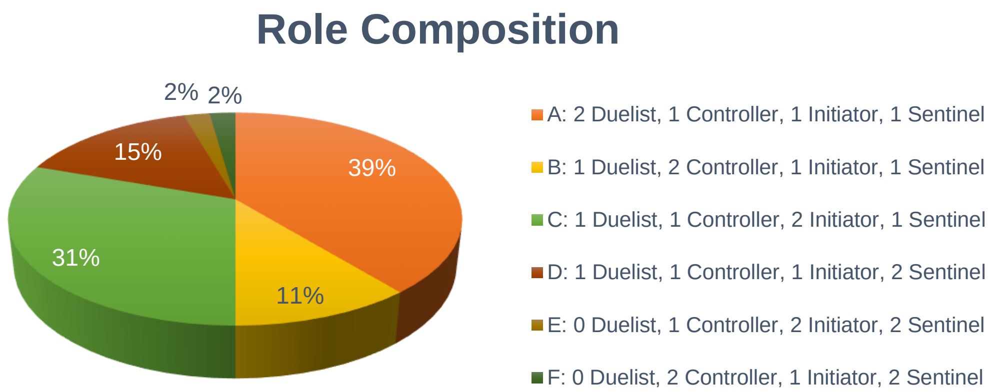


Figure 2. Survey Result

Based on Fig. 2, we will choose the top 3 role compositions: A, C, and D. To deepen the research, matches with a win rate under 50% will be chosen, and matches with a tie score will be excluded.

Table 1. Preferred role composition classification

Composition	Duelist	Controller	Initiator	Sentinel	Matches	Win-Lose	Win rate (%)
2 Duelist 1 Controller 1 Initiator 1 Sentinel	Jett Reyna	Astra	Skye	Cypher	2	1-1	50
1 Duelist 1 Controller 2 Initiator 1 Sentinel	Raze	Omen	Fade Breach	Chamber	11	7-4	63.63
	Raze	Astra	Fade Breach	Chamber	1	0-1	0
	Neon	Omen	Skye Breach	Chamber	1	1-0	100
	Neon	Omen	Fade Kay/O	Chamber	1	1-0	100
	Jett	Astra	Sova Breach	Killjoy	1	0-1	0
	Jett	Omen	Fade Breach	Chamber	1	0-1	0
	Jett	Omen	Fade Breach	Killjoy	4	1-3	25

1 Duelist 1 Controller 1 Initiator 2 Sentinel	Raze	Astra	Fade	Sage Chamber	1	Tied	
	Raze	Omen	Fade	Sage Chamber	1	1-0	100
	Phoenix	Omen	Fade	Chamber Killjoy	3	2-1	66.66

According to Table 1, there are four compositions with a win rate under 50%. Corresponding with the goal, teams that started as attackers and lost at half-match (shown in Appendix 1) will be selected as the object of research. The first composition includes Raze, Astra, Fade, Breach, and Chamber. FNC used this composition and lost to 100T. The second composition includes Jett, Astra, Sova, Breach, and Killjoy. BOOM used this composition and lost to ZETA. The third composition consists of Jett, Omen, Fade, Breach, and Killjoy. EDG used this composition and lost PRX. The fourth composition consists of Jett, Omen, Fade, Breach, and Killjoy. DRX used this composition thrice and lost thrice against three teams: LOUD, FPX, and OPTIC. According to the compositions, there are nine agents as the research variables: Raze, Jett, Astra, Omen, Sova, Fade, Breach, Chamber, and Killjoy.

2.2 Graph Construction

Each map in Valorant has callouts to help the players tell their teams where the enemies are located [28]. Based on previous information, we will assume that callout places and passable paths in Haven are nodes and links.

Table 1. Nodes labeling for the graph on Haven

Node	Callout Place	Node	Callout Place
A	Attacker Spawn	N	C Short
B	A Garden	O	C Site
C	Mid Window	P	A Site
D	Mid Doors	Q	A Back
E	C Lobby	R	B Site
F	A Lobby	S	C Window
G	Mid Courtyard (R)	T	A Main
H	Mid Courtyard (L)	U	A Main
I	C Long	V	A Link
J	C Cubby	W	B Back
K	A Long	X	C Link
L	A Short	Y	C Back
M	B Main	Z	Defender Spawn

Table 2. Edges labeling for the graph on Haven

Edge	Path Name	Edge	Path Name
a1	Attacker Spawn – A Garden	*a28*	B Back – B Site
a2	A Garden – A Lobby	*a29*	B Back – C Link
a3	A Lobby – A Long	*a30*	B Site – C Link
a4	A Lobby – A Short	*a31*	B Back – C Window
a5	A Long – A Site	*a32*	B Site – C Window
a6	A Short – A Site	*a33*	C Link – C Window
a7	A Site – A Back	*a34*	C Link – Defender Spawn
a8	A Back – A Site	*a35*	Attacker Spawn – Mid Doors
a9	A Site – A Tower	*a36*	A Garden – Mid Doors
a10	A Back – A Main	*a37*	Mid Window – Mid Doors
a11	A Tower – A Main	*a38*	Mid Doors – Mid Window
a12	A Tower – Defender Spawn	*a39*	Mid Doors – Mid Courtyard (L)
a13	A Main – Defender Spawn	*a40*	Mid Doors – C Short
a14	A Main – A Link	*a41*	Mid Courtyard (L) – C Short
a15	A Link – Defender Spawn	*a42*	C Short – C Window
a16	A Link – B Site	*a43*	C Short – C Site
a17	A Link – B Back	*a44*	Attacker Spawn – C Lobby
a18	Attacker Spawn – Mid Window	*a45*	A Garden – C Lobby
a19	A Garden – Mid Window	*a46*	Mid Window – C Lobby
a20	Mid Window – Mid Courtyard (R)	*a47*	Mid Doors – C Lobby
a21	Mid Window – Mid Courtyard (L)	*a48*	C Lobby – C Long
a22	Mid Courtyard (R) – Mid Courtyard (L)	*a49*	C Long – C Cubby
a23	Mid Courtyard (R) – B Site	*a50*	C Long – C Site
a24	Mid Courtyard (R) – B Main	*a51*	C Cubby – C Site
a25	Mid Courtyard (L) – B Main	*a52*	C Site – C Back
a26	B Main – B Site	*a53*	C Back – C Site
a27	B Site – B Back	*a54*	C Back – C Link

After labeling the nodes and edges, the initial graph (shown in the left frame of Figure 3) is connected. Dark blue nodes are callout places, and purple nodes are used explicitly for site callouts. The initial graph will be modified by eliminating paths that agents rarely or never go through. These are the assumptions for eliminating the tracks.

a) Each agent at least passed a path about four times in 12 rounds. Therefore, each team at least passed a path about 20 times in 12 rounds.
b) By doing cumulative computations for each team, the number of paths passed per agent does not affect the elimination. If the assumption is made per agent, a path will be simply eliminated because an agent rarely or never goes through it, even though that particular path is essential for other agents.

Table 3. Path elimination

Edge	Team				Total	Edge	Team				Total
	FNC	BOOM	EDG	DRX			FNC	BOOM	EDG	DRX	
a1	55	25	35	94	209	*a28*	0	0	2	5	7
a2	31	32	31	83	177	*a29*	0	0	0	2	2
a3	19	27	11	31	88	*a30*	0	0	2	0	2
a4	15	2	26	33	76	*a31*	0	0	0	0	0
a5	14	21	11	19	65	*a32*	3	0	0	0	3
a6	10	6	13	23	52	*a33*	2	0	2	0	4
a7	13	17	1	6	37	*a34*	0	2	0	0	2
a8	5	11	0	1	17	*a35*	10	15	0	52	77
a9	2	0	1	2	5	*a36*	17	10	1	36	64
a10	4	6	1	1	12	*a37*	8	4	0	2	14
a11	0	0	0	0	0	*a38*	1	4	3	4	12
a12	0	0	0	1	1	*a39*	10	18	5	36	69
a13	0	0	0	1	1	*a40*	9	3	9	34	55
a14	2	1	1	0	4	*a41*	5	6	2	9	22
a15	0	0	0	0	0	*a42*	5	5	1	5	16
a16	4	1	4	7	16	*a43*	6	7	9	38	60
a17	0	0	0	2	2	*a44*	13	12	27	37	89
a18	5	24	0	13	42	*a45*	6	7	21	13	47
a19	30	29	8	22	89	*a46*	1	2	1	0	4
a20	13	2	2	20	37	*a47*	12	14	12	5	43
a21	3	9	3	3	18	*a48*	23	30	29	55	137
a22	3	1	4	4	12	*a49*	15	9	0	21	45

a23	6	2	4	23	35	*a50*	1	13	25	34	73
a24	0	0	0	20	20	*a51*	8	8	0	16	32
a25	3	19	0	0	22	*a52*	7	3	4	18	32
a26	2	14	0	11	27	*a53*	3	2	1	6	12
a27	0	0	0	6	6	*a54*	3	1	5	4	13

Based on Table 3, 26 paths (grey) have been eliminated, and 28 paths (white) remain, which means only 28 paths will be used. The modified graph is shown in the right frame of Figure 3.

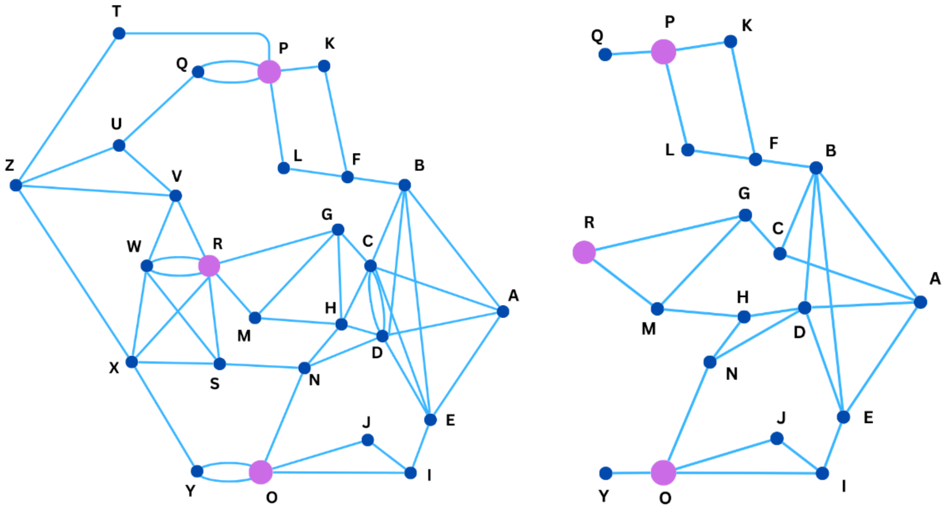


Figure 3. Initial (left) and modified (right) graph of Haven

Travel time on each path is slightly different depending on the weapons players use and how they pass the path. The weapon categories are primary, secondary, and melee. Most primary weapons are heavy, which causes players to move slower. Players will get faster as they switch to their secondary, or even better, their melee. Players must calculate their travel time by having 100 seconds to win the round. Since the travel time will be diverse, we did a simulation to get the average travel time.

Table 4. Average travel time

Edge	Time (s)	Edge	Time (s)	Edge	Time (s)	Edge	Time (s)
a1	8.7	*a8*	9.6	*a15*	8.8	*a22*	12.1
a2	3.7	*a9*	4.7	*a16*	8.3	*a23*	7.8
a3	8.3	*a10*	4.7	*a17*	4.2	*a24*	4
a4	4.8	*a11*	6.4	*a18*	6.8	*a25*	4.3
a5	4.9	*a12*	5.2	*a19*	5	*a26*	9.3
a6	6.7	*a13*	4.3	*a20*	6.7	*a27*	7
a7	3.9	*a14*	3.7	*a21*	8.8	*a28*	4.4

2.3 BIP Model

An integer programming in which each variable must equal 0 or 1 is called 0-1 integer programming or binary integer programming (BIP). This model represents an answer of yes or no in certain situation or option [29]. The following is the standard form of BIP model.

$$\text{Maximize } Z = \sum_{j=1}^{n} c_j x_j$$

$$\text{subject to: } \sum_{j=1}^{n} a_j x_j \le b,$$

$$x_j = 0 \text{ or } 1,\ j = 1, 2, ..., n$$

From the model, it has x_j as the decision variable. The other variables such as *aj*, *cj*, and *b* can have different meaning depends on the problem that needs to be solved by this model.

2.4 Data for Model

In order to construct the BIP model, some data from the game and match will be needed for model variables. Firstly, passable paths for agents and average travel time are obtained in Table 3 and Table 4. Next, each agent has a different number of times they pass through the path (shown in Appendix 2), which relates to their survivability. Agent survivability is based on how much the agent survives the paths and converts them into percentages and decimals. Lastly, each agent also has different skills obtained in a half-match. According to the selected teams, there are frequently used agents. Thus, we will sum up them based on the number of times each agent passes through the path, survivability, and obtained kills, respectively (shown in Appendix 2, 3, and 4).

3. Results and Discussion

3.1 BIP Model Construction

Notation for the model is defined as follows.

Parameters	
N	total of chosen agents
R	roles in Valorant
M	number of paths in graph
f_{hj}	frequency of h – th agent passing the j – th path
b_j	j – th path coefficient
k_{hj}	kill obtained by h – th agent via j – th path
s_{hj}	survivability of h – th agent passing the j – th path
t_j	travel time of the j – th path
B_{hi}	number of paths that must be passed by the h – th agent with i – th role
P_{hi}	number of kills obtained by the h – th agent with i – th role
L	survivability standard

W	maximum time for a round
Decision variable	
a_{hij}	1, if h – th agent with i – th role goes through the j – th path
	0, if h – th agent with i – th role does not go through the j – th path

Corresponding with the research goal, this post-match strategy maximizes the frequencies of agents passing through the paths by depending on the passable paths, obtained kills, survivability, and average travel time. By having the highest frequency, there is a possibility that the route consists of the most frequently passed paths. It does not eliminate the possibility that the route might have a low pass frequency. The objective function that maximizes the number of frequencies where agents pass through the paths is shown in Equation (1).

$$\text{Maximize } Z = \sum_{h=1}^{N} \sum_{i=1}^{R} \sum_{j=1}^{M} f_{hj} a_{hij} \tag{1}$$

The number of times agents pass through the path, shown in Appendix 2, will be used for f_{hj} and numbers for edges in graph.

The first constraint is passable paths, as shown in Table 3. Each path will be assumed to be passed once per round; agents will focus on entering the bombsite, planting the spike, and defending it until it blows up or the defender is eliminated. Considering the roles, agents have different paths that they must take. Duelist's agents must pass five paths, Initiator's and Controller's agents must pass four paths, and Sentinel's agents must pass three paths. The passable path constraint is shown in Equation (2).

$$\sum_{j=1}^{M} b_j a_{hij} = B_{hi}, \text{ for } b_j = 1 \tag{2}$$

Agents have different kills obtained in each path. Based on roles, each role has a different minimum kill. Duelist must eliminate the most, so it must have at least two kills. Initiator and Controller will help Duelist clear the path while Sentinel defends the planted spike and its team. Thus, those three roles must at least have one kill. The second constraint which is obtained kills constraint is shown in Equation (3).

$$\sum_{j=1}^{M} k_{hj} a_{hij} \geq P_{hi}, \text{ for } h = \{1, 2, ..., 9\} \tag{3}$$

Agents' obtained kills, shown in Appendix 3, will be used for k_{hj}.

The survivability of an agent depends on how much it survives the path. For frequently used agents, we will sum up the number of times they survived and divide by the number of uses. We assume that a path that never passes will have 100% survivability. The survivability constraint is shown in Equation (4).

$$\sum_{j=1}^{M} s_{hj} a_{hij} \geq L, \text{ for } h = \{1, 2, ..., 9\} \tag{4}$$

Each agent's survivability, shown in Appendix 4, will be used for s_{hj}.

The last constraint is about the travel time. As we mentioned before, each round has 100 seconds for the team and an extra 40 seconds of planted bomb to win the round. However, we will exclude the extra time with the assumption that the attackers will not have significant movement by the end of the route.

Therefore, the route planning must consider the time so the team will save time. The time travel constraint is shown in Equation (5).

$$\sum_{j=1}^{M} t_j a_{hij} \leq W \tag{5}$$

The average time travel shown in Table 4 will be used for t_j.

3.2 *Agents Routes*

With the help of POM QM for Windows in computation, each agent will have a different initial route. The routes are mainly made for one direction, and all the initial routes must be properly connected. Hence, we will make modified routes with additional constraints. The modified route with the highest Z will be chosen.

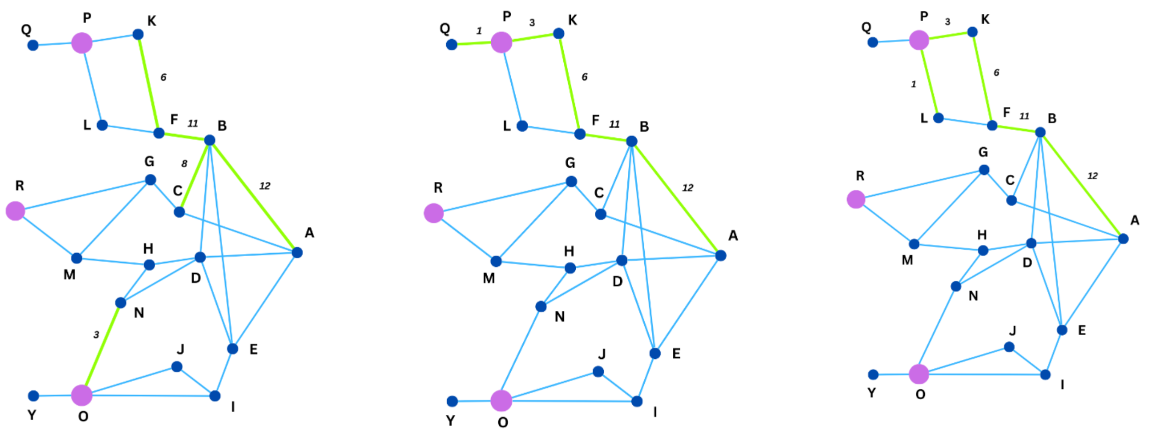
3.2.1 *Raze*

The initial route (Fig. 4(a)) has Z = 40. However, the route is not properly connected. As we can see, three edges, a1, a2, and a3 make their way to node P (A bombsite). Therefore, we will add additional constraints to connect all the chosen paths.

$$a_{(1,1,3)} + a_{(1,1,5)} + a_{(1,1,7)} = 3 \text{ that obtained } Z = 33, \text{ or} \tag{6}$$

$$a_{(1,1,3)} + a_{(1,1,5)} + a_{(1,1,6)} = 3 \text{ that obtained } Z = 33. \tag{7}$$

Based on Equations (6) and (7), Raze's Haven routes are directed and connected to A bombsite with one extended path for each route, which are P – Q for the first modified route (Fig. 4(b)) and P – L for the second modified route (Fig. 4(c)). The Raze player can choose the first or second modified routes because both Z have the same values.



(a) (b) (c)

Figure 4. (a) Raze initial route, (b) first, and (c) second modified routes

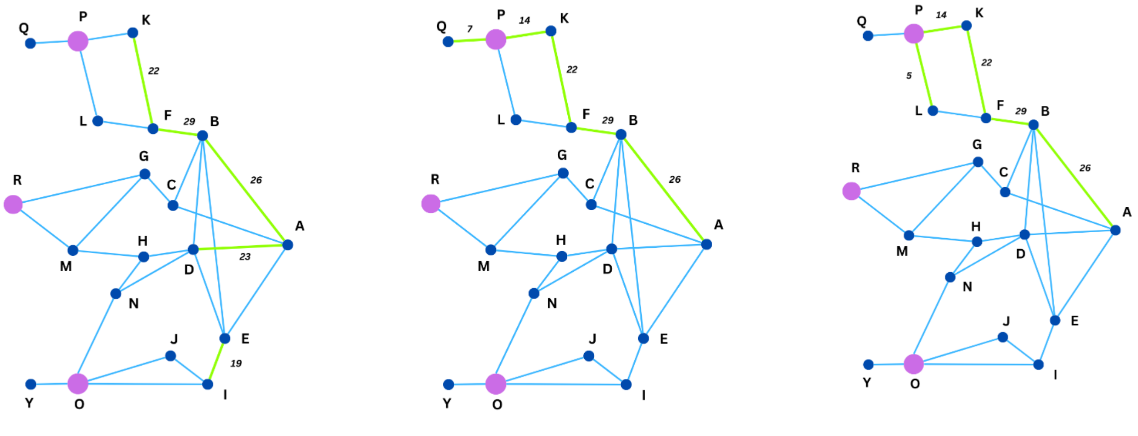
3.2.2 *Jett*

The second duelist agent has Z = 119 for its initial route. As shown in Fig. 4(a), the route is directed to node P (A bombsite), yet it still needs to be properly connected. Additional constraints will be added to connect the route.

$$a_{(2,1,3)} + a_{(2,1,5)} + a_{(2,1,7)} = 3 \text{ that obtained } Z = 98, \text{ or} \tag{8}$$

$$a_{(2,1,3)} + a_{(2,1,5)} + a_{(2,1,6)} = 3 \text{ that obtained } Z = 96. \tag{9}$$

Based on Equation (8) and (9), Jett's Haven routes are directed and connected to A bombsite with one extended path for each route, which are P – Q for the first modified route (Fig. 5(b)) and P – L for the second modified route (Fig. 5(c)). Seeing that Equation (8) has the highest Z, the players who use Jett will choose the first modified route (Fig. 5(b)).



(a) (b) (c)

Figure 5. (a) Jett initial route, (b) first, and (c) second modified routes

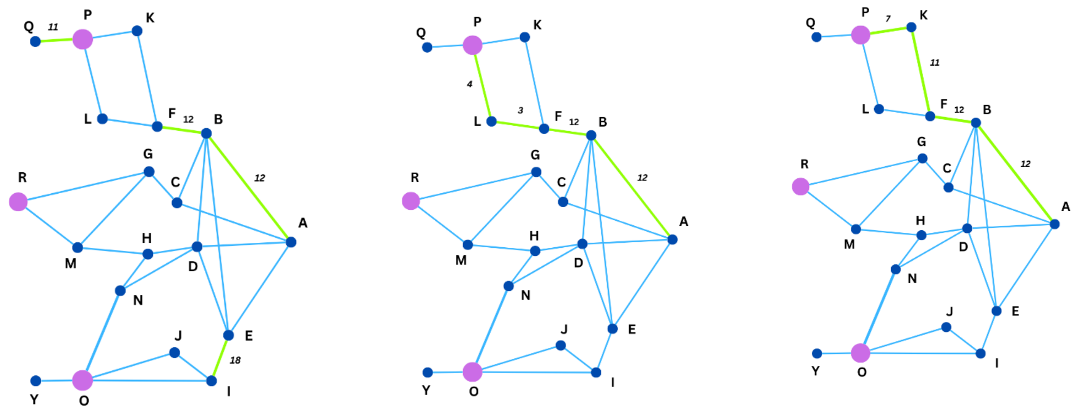
3.2.3 Astra

For its initial route (Fig. 6(a)), Astra has Z = 53, and the route is not properly connected. Since Controller must pass four paths, we will set back P – Q around K – P or L – P and transfer E – I around F – K or F – L.

$$\begin{cases} a_{(3,2,4)} + a_{(3,2,6)} = 2 \quad \text{that obtained Z = 31, or} \\ a_{(3,2,7)} + a_{(3,2,24)} = 0 \end{cases} \tag{10}$$

$$\begin{cases} a_{(3,2,3)} + a_{(3,2,5)} = 2 \quad \text{that obtained Z = 42.} \\ a_{(3,2,7)} + a_{(3,2,24)} = 0 \end{cases} \tag{11}$$

Based on Equation (10), the first modified route in Haven is directed to A bombsite by setting back P – Q to P – L and transferring E – I to F – L (Fig. 6(b)). Based on Equation (11), the second modified graph in Haven is directed to A bombsite by setting back P – Q to K – P and transferring E – I to F – K (Fig. 6(c)). Considering Equation (11) has the highest Z, the players who use Astra will choose the second modified route (Fig. 6(c)).



(a) (b) (c)

Figure 6. (a) Astra initial route, (b) first, and (c) second modified routes

3.2.4 Omen

Omen has Z = 97 for its initial route (Fig. 7(a)). The route must be properly connected, yet it is directed to two bombsites, A site (node P) and C site (node O). Additional constraints will be added to connect the route.

$$a_{(4,2,3)} + a_{(4,2,5)} = 2 \text{ that obtained } Z = 63, \tag{12}$$

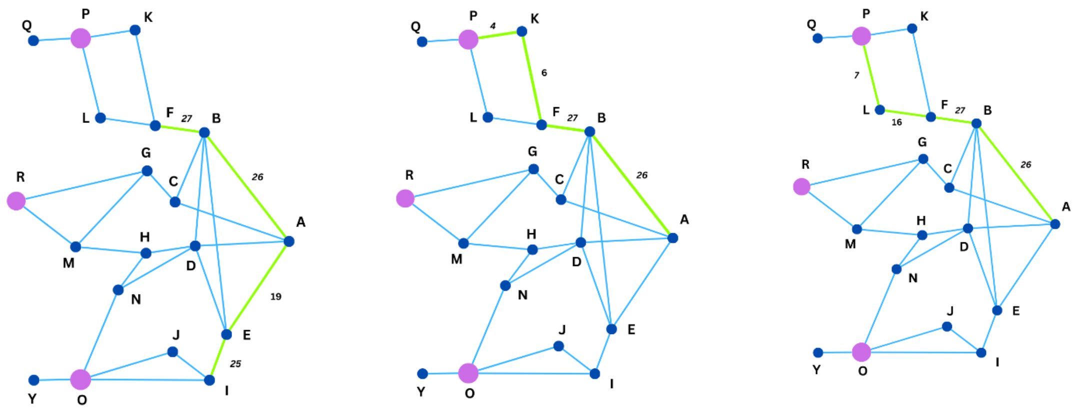
$$a_{(4,2,4)} + a_{(4,2,6)} = 2 \text{ that obtained } Z = 76, \tag{13}$$

$$\begin{cases} a_{(4,2,20)} + a_{(4,2,26)} = 2 \\ a_{(4,2,1)} + a_{(4,2,2)} = 0 \end{cases} \quad \text{that obtained Z = 62,} \tag{14}$$

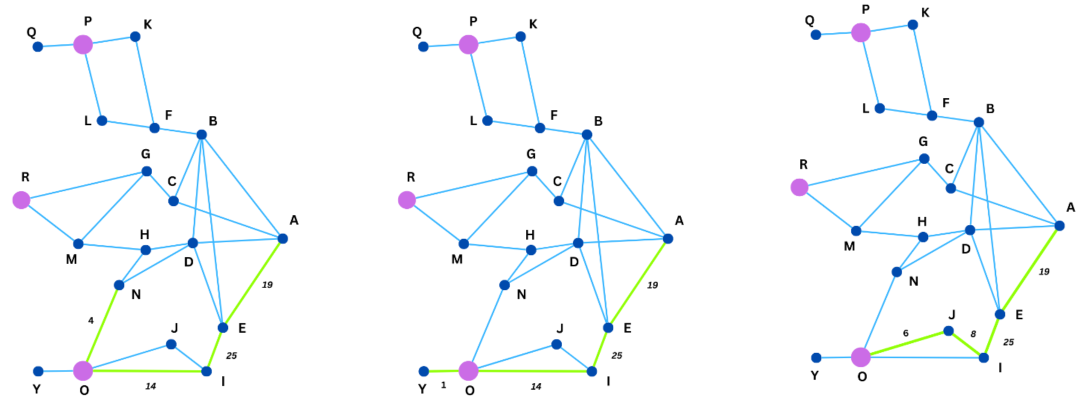
$$\begin{cases} a_{(4,2,26)} + a_{(4,2,28)} = 2 \\ a_{(4,2,1)} + a_{(4,2,2)} = 0 \end{cases} \quad \text{that obtained Z = 59, or} \tag{15}$$

$$\begin{cases} a_{(4,2,25)} + a_{(4,2,27)} = 2 \\ a_{(4,2,1)} + a_{(4,2,2)} = 0 \end{cases} \quad \text{that obtained Z = 58.} \tag{16}$$

Based on Equation (12) and (13), Omen's first and second modified routes are directed to A bombsite. The first modified routes transferred A – E and E – I to F – K and K – P (Fig. 7(b)). The second modified routes transferred A – E and E – I to F – L and L – P (Fig. 7(c)). Three other modified routes based on Equation (14), (15), and (16) are directed to C bombsite. The third one transferred A – B and B – F to I – O and O – N (Fig. 7(d)). The fourth modified route transferred A – B and B – F to I – O and O – Y (Fig. 7(e)). Lastly, the fifth modified route transferred A – B and B – F to I – J and J – O (Fig. 7(f)). Considering Equation (13) has the highest Z, the players who use Omen will choose the second modified route (Fig. 7(c)).



(a) (b) (c)



(d) (e) (f)

Figure 7. (a) Omen initial route, (b) first, (c) second, (d) third, (e) fourth, and (f) fifth modified routes

3.2.5 Sova

For its initial route, Z = 26 was obtained, but the route needed to be properly connected. Unfortunately, the route remains not properly connected even after additional constraints have been added because B – C or I – O cannot be moved together. Therefore, B – C and I – O are assumed to be optional paths that can be moved into two connected paths. A route directed to A bombsite will have F – K to K – P or F – L to L – P as options. B bombsite route will have C – G to G – R as an option. Lastly, options for C bombsite are combined with I – O. Hence, the options are A – E to E – I to I – O, A – E to E – I to O – Y, or A – D to D – N to N – O.

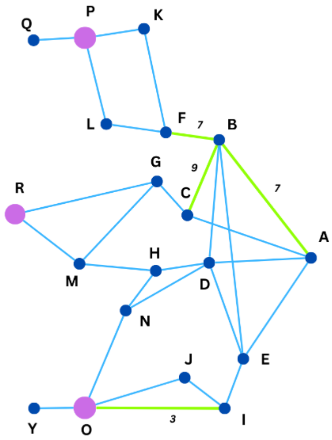
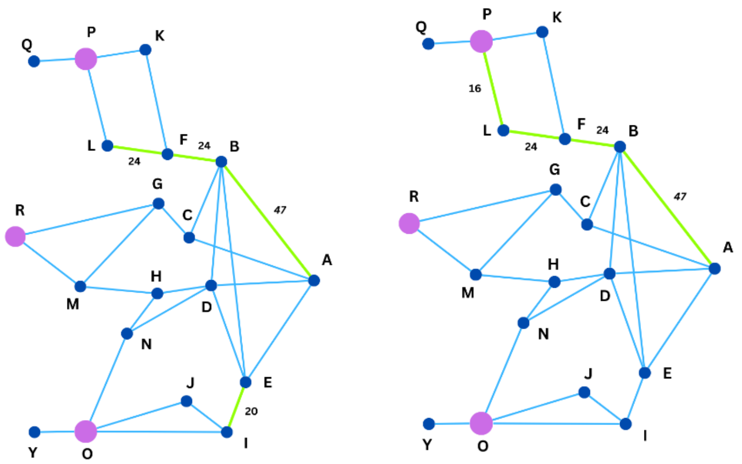


Figure 8. Sova Initial Route

3.2.6 Fade

The second Initiator agent has Z = 115 for its initial route. As shown in Fig. 9(a), the route is directed to node P (A bombsite) and almost properly connected. Thus, we only need to add a constraint to move E – I to L – P. The modified route will automatically be chosen for the players who use Fade.

$$a_{(6,3,6)} = 1 \text{ that obtained } Z = 111. \tag{17}$$



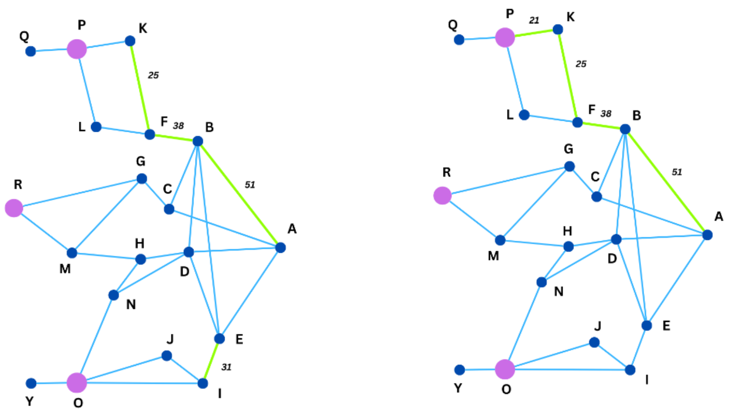
(a) (b)

Figure 9. (a) Fade Initial Route and (b) Modified Route

3.2.7 Breach

Breach has Z = 145 for its initial route. Similar to Fade, the route is directed to node P (A bombsite) and is almost properly connected. The difference is that Fade's initial route ends at F – L, while Breach ends at F – K. Thus, we only need to add a constraint to transfer E – I to K – P. The modified route will automatically be chosen for the players who use Breach.

$$a_{(7,3,3)} + a_{(7,3,5)} = 2 \text{ that obtained } Z = 135. \tag{18}$$



(a) (b)

Figure 10. (a) Breach initial and (b) modified routes

3.2.8 Chamber

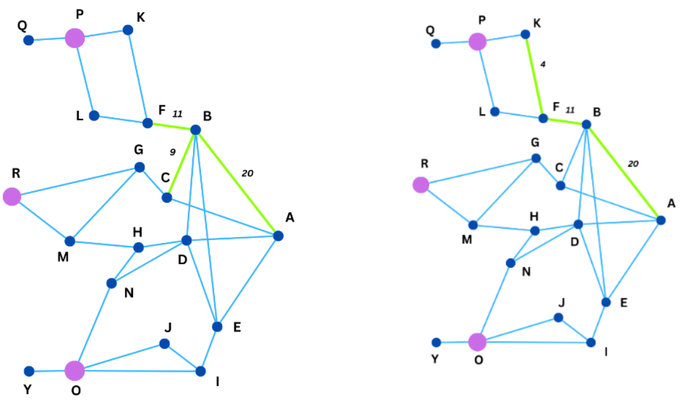
Chamber has Z = 40 for its initial route, yet it needs to be properly connected. As shown in Fig. 11(a), the route is directed to A bombsite and B bombsite. Additional constraints will be added to connect the route.

$$a_{(8,4,3)} = 1 \text{ that obtained } Z = 35, \tag{19}$$

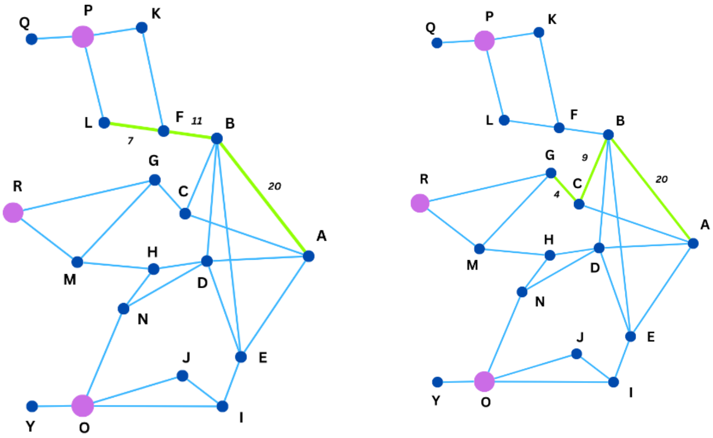
$$a_{(8,4,4)} = 1 \text{ that obtained } Z = 38, \text{ or} \tag{20}$$

$$a_{(8,4,9)} + a_{(8,4,10)} = 2 \text{ that obtained } Z = 33. \tag{21}$$

Based on Equation (19) and (20), Chamber's modified routes are directed and connected to A bombsite. The first modified route transferred B – C to F – K (Fig. 11(b)), while the second one transferred B – C to F – L (Fig. 11(c)). The third modified route is directed to B bombsite by transferring B – F to C – G (Fig. 11(b)). Since Equation (20) has the highest Z, the players who use Chamber will choose the second modified route (Fig. 11(c)). As a reminder, Chamber is a Sentinel, so entering the bombsite is not an obligation.



(a) (b)



(c) (d)

Figure 11. (a) Chamber Initial Route; (b) First, (c) Second, and (d) Third Modified Routes

3.2.9 Killjoy

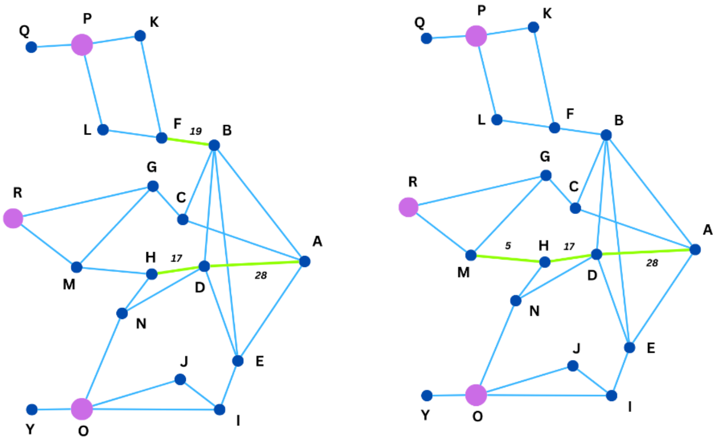
Killjoy has Z = 64 for its initial route. Based on Fig. 12(a), the route is directed either to the B or C sites. Additional constraints will be added to connect the route.

$$a_{(9,4,13)} + a_{(9,4,17)} = 2 \text{ that obtained } Z = 50, \tag{22}$$

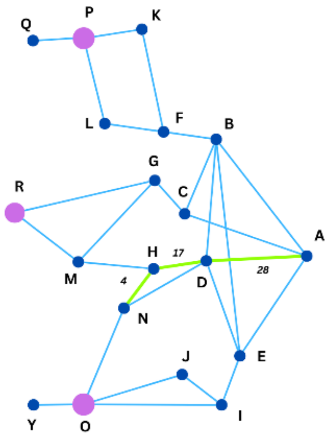
$$a_{(9,4,17)} + a_{(9,4,19)} = 2 \text{ that obtained } Z = 49, \text{ or} \tag{23}$$

$$a_{(9,4,16)} = 1 \text{ that obtained } Z = 59. \tag{24}$$

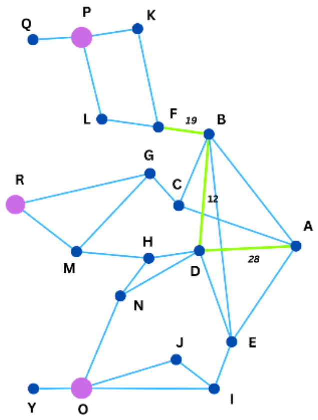
Based on Equation (22), Killjoy's first modified route is directed and connected to B bombsite by transferring B – F to H – M (Fig. 12(b)). Based on Equation (23), the second modified route is directed and connected to the C bombsite by transferring B – F to H – N (Fig. 12(c)). Lastly, the third modified route is directed and connected to A bombsite by transferring D – H to D – B (Fig. 12(d)). Considering that Equation (24) has the highest Z, the players who use Killjoy will choose the third modified route (Fig. 12(d)). As a reminder, Killjoy is a Sentinel, so entering the bombsite is not an obligation.



(a) (b)



(c)



(d)

Figure 12. (a) Killjoy Initial Route; (b) First, (c) Second, and (d) Third Modified Routes

4. Conclusions

This research developed a BIP model for the Haven attacker post-match strategy in Valorant. Excellent composition of roles and agents is necessary for a great strategy. The strategy is about generating optimal routes by maximizing the frequencies where agents pass through the paths that depend on the passable paths, obtained kills, survivability, and average travel time. We obtained one route for each chosen agent, except Raze and Sova. Raze has two modified routes with the same Z value, resulting in a player who uses Raze can choose the first or second modified routes. Sova's route cannot be made because the constraints cannot move one of the paths in the initial route. As an exception, there will be three route options to be played. Including Raze, 7 of 9 agents' routes are directed to A site. As for Killjoy's, the route is directed to the C site. The routes can be combined based on the preferred composition. This research does not guarantee an absolute win as the routes are another strategy option for the team.

Nonetheless, the best strategy is to select effective routes. Its effectiveness is measured based on the highest Z value for each agent. Also, the model for this strategy does not include each agent's skills.

Acknowledgement

This research was supported by the Faculty of Science and Mathematics, Diponegoro University for Student Research and Community Service with Project Code: 40.F4/UN7.F8/PP/II/2023. We are grateful to all the participants who have supported both time and effort to complete the research.

References

[1] J. Clement, "Most popular PC games based on monthly active users (MAU) worldwide in May 2023," statista. https://www.statista.com/statistics/1227532/top-pc-games-monthly-active-users/ (accessed Mar. 20, 2023)

[2] whatacoolwitch, "Be on Your Best Behaviour - VALORANT Support," Riot Games. https://support-valorant.riotgames.com/hc/en-us/articles/360044270174-Be-on-Your-Best-Behavior (accessed Feb. 12, 2024)

[3] A. Goslin, "Valorant: Everything we know about Riot Games' new shooter," Polygon. https://www.polygon.com/2020/3/2/21155158/valorant-project-a-riot-games-shooter (accessed Jan. 12, 2023)

[4] Enzo Zalamea, "A Guide to Valorant Competitive Mode," DiamondLobby. https://diamondlobby.com/valorant/ranked-mode-valorant/ (accessed Feb. 12, 2024)

[5] Valorant Wiki, "Competitive | Valorant Wiki | Fandom," Valorant Wiki. https://valorant.fandom.com/wiki/Competitive (accessed Mar. 01, 2024)

[6] R. Wilcox *et al.*, "Valorant Economy Guide," IGN. https://www.ign.com/wikis/valorant/Valorant_Economy_Guide (accessed Feb. 10, 2024)

[7] Si Yan, "How Long Is A Valorant Game in Each Mode (2023)," VideoGamer. https://www.videogamer.com/guides/how-long-is-a-valorant-game/ (accessed Feb. 10, 2024)

[8] playvalorant, "VALORANT: Riot Games' competitive 5v5 character-based tactical shooter." https://playvalorant.com/en-us/ (accessed Jan. 12, 2023)

[9] Valorant Wiki, "Agents | Valorant Wiki | Fandom," Valorant Wiki. https://valorant.fandom.com/wiki/Agents (accessed Jan. 02, 2023)

[10] J. Heath, "All VALORANT classes explained," Dot Esports. https://dotesports.com/valorant/news/all-valorant-classes-explained (accessed Jan. 02, 2023)

[11] S. Gunpinar and G. Centeno, "An integer programming approach to the bloodmobile routing problem," *Transp Res E Logist Transp Rev*, vol. 86, pp. 94–115, Feb. 2016, doi: 10.1016/j.tre.2015.12.005.

[12] S. Wang, W. Zhang, Y. Bie, K. Wang, and A. Diabat, "Mixed-integer second-order cone programming model for bus route clustering problem," *Transp Res Part C Emerg Technol*, vol. 102, pp. 351–369, May 2019, doi: 10.1016/j.trc.2019.03.019.

[13] M. J. Mauricio, P. J. de Rezende, and C. C. de Souza, "Solving the geometric firefighter routing problem via integer programming," *Eur J Oper Res*, vol. 274, no. 3, pp. 1090–1101, May 2019, doi: 10.1016/J.EJOR.2018.10.037.

[14] Q. Xue, X. Yang, J. Wu, H. Sun, H. Yin, and Y. Qu, "Urban rail timetable optimization to improve operational efficiency with flexible routing plans: A nonlinear integer programming model," *Sustainability (Switzerland)*, vol. 11, no. 13, 2019, doi: 10.3390/su11133701.

[15] C. Ntakolia and D. K. Iakovidis, "A route planning framework for smart wearable assistive navigation systems," *SN Appl Sci*, vol. 3, no. 1, 2021, doi: 10.1007/s42452-020-04038-6.

[16] Z. Huang, W. Huang, and F. Guo, "Integrated sustainable planning of micro-hub network with mixed routing strategy," *Comput Ind Eng*, vol. 149, 2020, doi: 10.1016/j.cie.2020.106872.

[17] C. Iliopoulou and K. Kepaptsoglou, "Integrated transit route network design and infrastructure planning for on-line electric vehicles," *Transp Res D Transp Environ*, vol. 77, pp. 178–197, Dec. 2019, doi: 10.1016/J.TRD.2019.10.016.

[18] K. Balaji and M. Rabiei, "Carbon dioxide pipeline route optimization for carbon capture, utilization, and storage: A case study for North-Central USA," *Sustainable Energy Technologies and Assessments*, vol. 51, 2022, doi: 10.1016/j.seta.2021.101900.

[19] X. Li, D. Li, X. Hu, Z. Yan, and Y. Wang, "Optimizing train frequencies and train routing with simultaneous passenger assignment in high-speed railway network," *Comput Ind Eng*, vol. 148, 2020, doi: 10.1016/j.cie.2020.106650.

[20] L. Wang, S. Gao, K. Wang, T. Li, L. Li, and Z. Chen, "Time-Dependent Electric Vehicle Routing Problem with Time Windows and Path Flexibility," *J Adv Transp*, vol. 2020, 2020, doi: 10.1155/2020/3030197.

[21] W. Liu *et al.*, "A hybrid ACS-VTM algorithm for the vehicle routing problem with simultaneous delivery & pickup and real-time traffic condition," *Comput Ind Eng*, vol. 162, 2021, doi: 10.1016/j.cie.2021.107747.

[22] J. X. Cao, X. Wang, and J. Gao, "A two-echelon location-routing problem for biomass logistics systems," *Biosyst Eng*, vol. 202, 2021, doi: 10.1016/j.biosystemseng.2020.12.007.

[23] S. K. Seo, Y. G. Yoon, J. sung Lee, J. Na, and C. J. Lee, "Deep Neural Network-based Optimization Framework for Safety Evacuation Route during Toxic Gas Leak Incidents," *Reliab Eng Syst Saf*, vol. 218, p. 108102, Feb. 2022, doi: 10.1016/J.RESS.2021.108102.

[24] K. F. Krutein and A. Goodchild, "The isolated community evacuation problem with mixed integer programming," *Transp Res E Logist Transp Rev*, vol. 161, 2022, doi: 10.1016/j.tre.2022.102710.

[25] F. K. Asare, "Application of Integer Programming for Mine Evacuation Modeling With Multiple Transportation Modes," Master of Science, University of Nevada, Reno, 2023. https://www.proquest.com/openview/1b0e1fa907603ab449b4a6e2e5ea8f38/1?pq-origsite=gscholar&cbl=18750&diss=y (accessed May 10, 2023)

[26] C. Chen, P. Yang, and S. Liu, "The Mission Planning of Multi-wave Missile Launching Based on 0-1 Integer Programming," in *2020 International Conference on Artificial Intelligence and Communication Technology (AICT 2020)*, 2020. doi: 10.23977/AICT2020047.

[27] F. F. Fibriyanto, "Penentuan Rute Rotasi Tercepat Dalam Map Breeze Gim Valorant Menggunakan Algoritma Dijkstra Makalah," 2021. [Online]. https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2021-2022/Makalah2021/Makalah-Matdis-2021%20(59).pdf (accessed Sep. 24, 2023)

[28] R. Wilcox *et al.*, "Valorant Callouts," IGN. https://www.ign.com/wikis/valorant/Valorant_Callouts (accessed Feb. 12, 2024)

[29] W. L. Winston, *Operations Research: Applications and Algorithms*, 4th ed. Belmont: Brooks/Cole, 2004.

Appendix

1. Table of VCT 2022 (only first half-matches in Haven)

No.	Match	Team	Side	Result	No.	Match	Team	Side	Result
1.	PRX vs EDG	PRX	Defender	Won	9.	TL vs PRX	TL	Defender	Won
		EDG	Attacker	Lost			PRX	Attacker	Lost
2.	LEV vs TL	LEV	Defender	Won	10.	LOUD vs LEV	LOUD	Defender	Won
		TL	Attacker	Lost			LEV	Attacker	Lost
3.	LEV vs PRX	LEV	Defender	Tie	11.	TL vs FNC	TL	Defender	Lost
		PRX	Attacker	Tie			FNC	Attacker	Won
4.	DRX vs FUR	DRX	Defender	Won	12.	LOUD vs DRX	LOUD	Defender	Won
		FUR	Attacker	Lost			DRX	Attacker	Lost
5.	100T vs FNC	100T	Defender	Won	13.	OPTIC vs XSET	OPTIC	Defender	Lost
		FNC	Attacker	Lost			XSET	Attacker	Won
6.	ZETA vs BOOM	ZETA	Defender	Won	14.	FPX vs DRX	FPX	Defender	Won
		BOOM	Attacker	Lost			DRX	Attacker	Lost
7.	KRÜ vs XIA	KRÜ	Defender	Lost	15.	OPTIC vs DRX	OPTIC	Defender	Won
		XIA	Attacker	Won			DRX	Attacker	Lost
8.	FUR vs FNC	FUR	Defender	Tie	16.	LOUD vs OPTIC	LOUD	Defender	Won
		FNC	Attacker	Tie			OPTIC	Attacker	Lost

2. Table of agents' frequencies in passing through the paths

Node	Agents								
	Raze	Jett	Astra	Omen	Sova	Fade	Breach	Chamber	Killjoy
a1	12	26	12	26	7	47	51	20	8
a2	11	29	12	27	7	24	38	11	19
a3	6	22	11	6	4	2	25	4	8
a4	2	4	3	16	1	24	12	7	7
a5	3	14	7	4	4	2	21	4	6
a6	1	5	4	7	3	16	5	5	6
a7	1	7	11	1	1	1	10	2	3
a8	0	5	2	0	7	2	12	1	13
a9	8	12	10	0	9	12	13	9	16

a10	2	7	2	0	1	9	9	4	3
a11	1	8	1	6	1	8	5	2	3
a12	0	4	0	3	0	1	2	0	10
a13	0	6	1	0	2	1	6	1	5
a14	1	6	1	3	1	2	8	0	5
a15	5	23	3	5	1	7	5	2	28
a16	5	5	6	2	1	17	11	5	12
a17	3	14	6	10	1	6	12	1	17
a18	1	8	3	4	1	16	8	6	9
a19	0	6	5	0	0	3	4	0	4
a20	3	10	2	4	1	14	11	2	13
a21	4	9	11	19	0	16	17	5	6
a22	1	6	3	8	1	6	13	8	1
a23	3	5	8	3	1	6	10	4	2
a24	5	19	18	25	4	20	31	8	7
a25	4	3	1	8	2	6	16	4	1
a26	0	13	5	14	3	15	8	5	10
a27	3	1	2	6	2	1	12	2	3
a28	2	10	0	1	0	4	7	3	5

3. Table of agents' obtained kills

Node	Agents								
	Raze	Jett	Astra	Omen	Sova	Fade	Breach	Chamber	Killjoy
a1	0	0	0	0	0	0	0	1	0
a2	0	1	0	1	0	2	0	1	2
a3	0	2	0	1	0	0	3	1	0
a4	0	1	0	2	0	0	0	0	0
a5	1	3	2	1	0	0	6	0	0
a6	1	1	1	4	0	3	0	1	4
a7	2	3	1	0	0	1	0	0	0
a8	0	0	0	0	0	0	0	0	5

a9	0	0	0	0	0	0	1	0	0
a10	0	0	0	0	0	0	0	0	0
a11	1	1	0	3	0	2	0	0	0
a12	0	0	0	0	0	0	0	0	0
a13	0	0	0	0	0	0	1	0	0
a14	0	1	0	0	0	0	1	0	1
a15	0	1	0	0	0	0	0	0	1
a16	0	0	0	0	0	1	0	1	0
a17	0	0	0	0	0	1	0	2	1
a18	0	0	0	0	0	2	0	2	0
a19	0	0	0	0	0	0	0	0	0
a20	3	3	0	0	0	4	6	1	9
a21	0	0	0	0	0	0	0	0	0
a22	0	0	0	0	0	1	0	0	0
a23	0	0	0	1	0	0	0	0	0
a24	1	2	0	1	0	0	0	3	0
a25	0	0	0	1	0	0	2	0	0
a26	0	9	0	5	3	1	2	1	2
a27	0	0	0	3	0	0	0	0	1
a28	1	1	0	1	0	1	0	1	3

4. Table of agents' survivabilites

Node	Agents								
	Raze	Jett	Astra	Omen	Sova	Fade	Breach	Chamber	Killjoy
a1	1	1	1	1	1	1	1	0.95	1
a2	0.91	1	0.917	0.90625	0.857	0.95	0.91076	1	0.8875
a3	0.93	0.62	1	0.5	1	1	0.902966	1	0.75
a4	1	1	1	0.8125	1	0.9444	1	0.5665	1
a5	0.67	0.66	0.857	0.9375	1	0.8	0.711066	0.75	0.65
a6	0	0.76	0.5	0.6875	0.333	0.3534	0.9168	0.8335	0.625
a7	0	0.766	0.5	1	0.6	1	0.7167	0.75	1

a8	1	1	1	1	1	1	0.9168	1	1
a9	1	1	0.9	1	1	1	1	1	1
a10	1	1	1	1	0	0.95	1	1	0.75
a11	1	0.8	1	0.91675	1	0.5334	0.7501	0.75	1
a12	1	1	1	1	1	1	1	1	0.95825
a13	1	0.966	1	1	1	1	0.88893	1	1
a14	0	0.96	1	0.875	1	1	0.8751	1	0.625
a15	1	0.9714	1	1	1	1	1	1	1
a16	1	1	1	1	1	1	1	0.75	0.91675
a17	1	1	0.833	1	1	0.9334	0.9168	1	1
a18	1	1	1	0.9375	1	0.8934	1	1	0.91675
a19	1	0.95	1	1	1	1	1	1	1
a20	0.67	0.5666	0	0.41675	1	0.87	0.444466	0.5	0.79175
a21	1	1	1	1	1	1	1	1	1
a22	1	1	1	1	1	1	0.9168	1	1
a23	1	0.9	1	1	1	1	0.9335	1	1
a24	0.6	1	0.944	0.8275	1	0.9	0.97633	0.75	1
a25	1	0.8666	1	0.8125	0.5	1	0.8334	1	1
a26	1	0.55	0.6	0.63675	0.333	0.6248	0.8501	0.857	0.5
a27	1	0.8	0.5	1	0.5	0.8	0.8056	0.75	1
a28	0.5	0.6	1	0.75	1	0.8	0.8335	1	0.75