

תוכן עניינים

תקציר	4
מבוא	4
תיאור המערכת הנחקרת	4
מטרות	5
מודל הסימולציה	6
תיאור המודל	6
הנחות	7
ניתוח מצב קיימן והצעת חלופות לשיפור מצב קיימן	7
בחירת מדרדים לניתוח	7
סוג המערכת	8
ניתוח מצב קיימן ביחס לממדים ולמטרות	9
בחירת חלופות	10
השוואה סטטיסטית בין מצב קיימן לחלופות מוצעות	12
מסקנות	12
נספחים	13
נספח 1 – מודל הסימולציה	13
נספח 1.1 – מסלול ראשי	13
נספח 1.2 – מסלול טריגר	13
נספח 1.3 – דוגמא לאחד ממסלולי דון האוכל	13
נספח 1.4 – מסלול בר חיצוני	14
נספח 1.5 – מסלול קינוחים וריקודים	14
נספח 1.6 – תיאור כל המסלולים שהישיות עוברות	14
נספח 2 – בדיקת נכונות המודל	18
נספח 3 – התפלגיות סימולציה, תוכנות ומשאים	21
נספח 3.1 – זמני הגעה	21
נספח 3.2 – זמני שירות	22
נספח 3.3 – זמני פעולה	22
נספח 3.4 – טבלת ישוות	22
נספח 3.5 – טבלת תוכנות	23
נספח 3.6 – קביעת רמת שבוע של קבוצה	23
נספח 3.7 – קביעת התפלגות הגעת הזוגות והמשפחה	24

27.....	נספח 4 – חישוב המדדים שבחרנו
27.....	נספח 4.1 – חישוב זמן המתנה ממוצע בעמדות קבלת הפנים
27.....	נספח 4.2 – חישוב אחוז המאוכזבים באירוע
28.....	נספח 4.3 – חישוב אחוז הנוכחים בחופה
28.....	נספח 5 – חישוב מספר החזרות הנדרש להרצת הסימולציה
28.....	נספח 5.1 – חישוב הממצאים ווטיות התקן
29.....	נספח 5.2 – חישוב הדיק היחסי ומספר החזרות הנדרש
30.....	נספח 5.3 – בדיקה שמספר החזרות הנדרש אכן עומד בדיק היחסי
31.....	נספח 5.4 – תוצאות המדדים לאחר הרצת מספר החזרות הנדרש
31.....	נספח 5.5 – השוואת המודלים (מבחני χ^2 מזוגים)
31-44.....	נספח 6 – קוד הסימולציה

3. תקציר

בפרויקט שביצענו התבקשנו למדל באמצעות תוכנת סיד-ר, סימולציה המדמה התנהלות אירוע חתונה באולם אירועים "הבראר של שבע". תחילת התבקשנו לבחון את פעילותם אולם האירועים במצב הקיימים ולאחר מכן להמליץ על שיפורים אפשריים מהתוך רשות חלופות נתונה.

לאחר מידול והרצת מודל המערכת נחקרו ונקבעו ממדדים אשר לפיהם בחנו את שיפור המערכת בהתאם למטרות הפרויקט. בהינתן תקציב של **50,000** ש' המודדים שנבחרו הינם: זמן המתנה ממוצע לכל עמדות דוכני האוכל בקבלת הפנים (נרצה למשך), אחוז האורחים שייצאו מאוכזבים – לאכול קינוח או הגיעו אליו (נרצה למשך), אחוז האורחים שהגיעו לחופה לפני שנגמרה (נרצה למשך). מניתוח המצב הקיימים התגלה כי ערך הזמן של זמן המתנה ממוצע לעמדות הינו **21.37 دق'**. ערך הזמן של אחוז האורחים שייצאו מאוכזבים הינו **26.5%**. ערך הזמן של אחוז האורחים שהגיעו לחופה לפני סיום הינו **27.61%**.

על מנת לשפר את ערכי המודדים במצב הקיימים, בחרנו 2 חלופות אפשריות אשר הערכנו כי יתרמו לשיפור ערכי ממדדים אלו. בחופה הראשונה בחרנו להוסיף עמדת פירוט שתפתח לדקה וכל אורחי האולם השוהים יכולו לקחת ממנה, בתוספת של הויסת 2 מצלרים – לעמודת הסושי עם המלצר הבודד ולעמודת הטורטיה. העלות הכוללת לכך תהיה 50,000 ש'. החופה השנייה שבחרנו היא הוספה לעמודת טורטיה נוספת לעמדת הקיימת והוספה 2 מצלרים – אחד לסייע עם המלצר הבודד ולעמודת הצמחוני. העלות הכוללת לכך תהיה 45,000 ש'. לשם מציאת החופה הטובה מבניהם, נדרשנו לבדוק באמצעות נוסחאות הדיק היחס והדיק הוגש מהו מספר הריצות הנדרש, וקיים שיש צורך ב-**30** ריצות שונות. ביצענו מבחני \pm מזוגים להשוואה בין חלופות (בר"מ כללית של **9%**).

בהתאם לתוצאות ולניתוחים הסטטיסטיים שערכנו, אנו ממליצים לבעל האולם על **חלופה מס' 2**, מאחר והוא גורמת לשיפור גדול יותר מחלופה 1. באופן כללי, 2 החולופות טובות יותר מה מצב הקיימים בכלל המודדים, אך בהשוואה בהם ניתן לראות יתרון ברור במידד אחוז הנוכחים בחופה לחלופה 2. זמן המתנה ממוצע בתור הוא **14.38 دق'**, ממד אחוז המאוכזבים הינו **24%** ומדד אחוז הנוכחים בחופה **45%**.

4. מבוא

4.1. תיאור המערכת הנחקרת

המערכת אותה אנו חוקרם היא התנהלות אירועי אירועים "הבראר של שבע". אירוע חתונה מתקיים בין שעות 18:30 ל- 00:30. מבנה אירוע החתונה באולם הוא קבוע – קבלת פנים הכוללת עמדות אוכל שונות, לאחר מכן מתקיימת החופה שבוסף האורחים נכנסים לאולם. בתוך האולם מוגשות המנות העיקריות ומתקיימת מסיבת הריקודים.

באירוע שלנו מגיעים אורחים בכמותות שונות – משפחות, זוגות ויחידים. המשפחות והזוגות מגיעות ברכבים פרטיים ואילו היחידים מגיעים בהסעות. קבצי הגעת המשפחות, הזוגות והיחידים ומספר הנפשות במשפחה מוצגים בטבלת התפלגות ([3.1 נספח](#)). קביעת התפלגות הגעת המשפחות והזוגות מפורט כאן ([נספח 3.7](#))

תחילה האורחים יעברו דרך 2 עמדות קבלת שם יקבלו מס' שולחן. לאחר מכן, בקבלת הפנים יפגשו האורחים בעמדת פוקאצ'ות (3 טבחים), עמדת באנים (3 טבחים), 2 עמדות סושי (עמדה עם 2 טבחים ועמדה עם טבח אחד), עמדת טורטיות ועמדה צמחונית גדולה (שניהם עם 4 טבחים). בין עמדות האוכל ממוקם בר חיצוני (3 ברמנים) בו יוכל האורחים להתכבד במשקאות איכוטיים. זמן השירות של העמדות השונות ([נספח 3.2](#)), זמני אכילת המנות ([נספח 3.3](#)) ובחרית מספר השולחן ([נספח 3.4](#)) מפורטים בנספחים הנלוויים.

לאחר קבלת הפנים, תחילת החופה בין שעה 20:50 ל- 21:10. ברגע ההכרזה על החופה, הטבחים יסימו להכין את המנות עבור האורחים שמתינו בתור ולאחר מכן יסגרו את העמדות. כל האורחים יעברו למתחם החופה עד לסיוםה. זמני התפלגות תחילת החופה ואורכה מוצגים בנספח הנלווה. ([נספח 3.4](#))

לאחר סיום החופה יעברו האורחים לאולם המרכזים. יחידים וחוגות ייגשו תחיליה לשני הברים הפנימיים (בר אחד עם 5 ברמנים ואחד עם 7 ברמנים) כדי לקבל משקה ולאחר מכן יחפשו את שולחנם ויתישבו בו. משפחות יחפשו תחיליה את השולחן ויתישבו בו. בשולחן, יקבלו האורחים מנת עיקרית (צמחונית או בשנית). הקבוצות ישארו לשבת בשולחן עד שבן המשפחה האחרון מסיים לאכול את המנה שלו. זמני השירות בבר הפנימי ([נספח 3.2](#)), זמני חיפוש השולחנות, זמני אכילת המנות ([נספח 3.3](#)) מוצגים בנספחים הנלוויים.

לאחר האוכל, יילכו האורחים לעמדת הקינוחים. באזור הקינוחים יכולים לשחות 15 קבוצות קבועות אשר רמת השבוע שלה גובהה (מעל 4.5) תוויתר על מקומה בתור לקבוצה עם רמת שבוע נמוכה יותר (מתחת ל 4.5). קבוצה אשר לא היה עבורה מקום באזור הקינוחים, תרകוד זמן מה ותנסה לגשת שוב לעמדת הקינוחים. במידה וудין לא היה מקום בעמדת הקינוחים, הקבוצה תלך לרקוד ולאחר מכן תנסה שוב. במידה ולא הצליחה, הקבוצה תישאר מאוכזבת מהאירוע ותעזוב את האירוע מידית. זמני אכילת הקינוחים, זמני הריקודים ([נספח 3.3](#)) וקביעת רמת השבוע של קבוצה ([נספח 3.6](#)) מוצגים בנספחים הנלוויים.

4.2. מטרות

מטרת אולם האירועים היא ליצור חוות שירות מיטבית עבור האורחים והזוג שמתחתקן. המטרה העיקרית של הפרויקט הינה "על התנהלות אולם האירועים, התנהלות נכונה ושיפור התהילה השוטף באולם "הбар של שבע". מטרת הסימולציה היא לבחון את המצב הנוכחי של התנהלות אולם האירועים ולמצוא נקודות בהם נוכל לשפר את המדרדים שלנו ע"י מציאת חלופות.

5. מודל סימולציה

5.1. תיאור המודל

המודל מחלק למשתמשים מסלולים עיקריים:
(הסבירים על הישיות השונות – [נספח 3.4](#) ותכונותיהם – [נספח 3.5](#))
מסלול ראשי (mainPath):

מסלול הראשי של אורחים המגיעים לחנותה. כל ישות של זוג או משפחה מגיעה למסלול זה לאחר תהליך של החניה וישות של יחיד מגיעה למסלול זה ישירות לאחר הגעת האוטובוסים. עם הגיעו למסלול זה נקבעות תכונות ראשוניות לשינויים שחשיבותם לסימולציה (סדר ההליכה לדוכני האוכל, type). מתוך מסלול זה נשלחות הישיות השונות אל המסלולים השונים: עדמות הקבלה, דוכני האוכל, עדמת הבר הפנימי, אכילת המנה העיקרית, עדמת הקינוחים ורחבת הריקודים. ([נספח 1.1](#))

מסלולי טריגרים (trigger, reserPath):

ישנים 2 מסלולי טריגרים אשר לכל אחד ייעוד שונה. Trigger – מטרתו למדל את הגעת האוטובוסים עם היחידים אל האולם. הטריגר מבצע אקטיבציה למסוג generator מסווג יחיד כך שבבאת אחת נוצרים כל הישיות מסווג יחיד. Reset – מטרתו לבצע את כל הפעולות שמצריכות זמן במהלך החנותה – סגירת הבר החיצוני, סגירת דוכני האוכל, הגדרת משטני סימולציה בתחילת כל הרצת סימולציה חדשה. ([נספח 1.2](#))

מסלול עדמות הקבלה (reception):

לאחר הגיעו למסלול הראשי, כל הישיות עוברות דרך אחת מתוך שתי עדמות הקבלה שקיימות ומתקבלות את מספר השולחן שלהם. בחירת האורחים לאיזה עדמה תלכת הינה לפי מדיניות התווך הקצר ביותר (בנייה) לדמות את המזינות בצורה הטובה ביותר, לפי ראות עינינו). מתוך מסלול זה, 7% מהאורחים מבינים שלא הגיעו לאולם הנכון ולכן עוזבים את האירוע ולא חוזרים למסלול הראשי. אלו שאינם עוזבים, מקבלים את מספר השולחן שלהם לפי ההתפלגות המתאימה.

מסלול דוכני האוכל (sushiPath, tortillasPath, veggiesPath, focacciaPath, bunsPath):

הישיות שמשיטים בעמדות הקבלה מתקדמות לאחר ביצוע ה-(clone) כל אחד בסדר שונה לדוכני האוכל, "פ' התכונות של כל ישות שמוגדרת רק לאחר ביצוע ה-(clone) וזה כדי שלכל אורח עצמאי יהיה מסלול שונה כפי שהוא. מתוך כל אחד ממסלולים אלו, בהסתברות מתאימה לשיטת האורח יכול לבר החיצוני. הוספנו את ההליכה לבר החיצוני מתוך כל אחד ממסלולי האוכל על מנת שנוכל למדל את העבודה שהיחידים אשר מגיעים מאוחר מלהזנות והמשפחות, הולכים רק ל-3 עדמות. על כן, כאשר מוגדר להם במסלול הראשי את סדר ההליכה לדוכני האוכל, מתווספים להם שני עמדות פיקטיביות שלא מbezחות להן זמן, כדי שנוכל להשתמש בפונקציית () rollback ביצוע המעבר לדוכני האוכל. ([נספח 1.3](#))

מסלול בר פנימי (internalBarPath):

לאחר סיום החופה זוגות יחידים הולכים לשתו בבר הפנימי מתוך מסלול הראשי. בחירת הבר הפנימי מתוך 2 הברים האפשריים מתבצעת לפי מדיניות התווך הקצר ביותר. בסיום קבלת המשקה האורחים חוזרים בחזרה למסלול הראשי לקבלת המנה העיקרית. ([נספח 1.4](#))

מסלול קינוחים (dessertPath):

לאחר סיום אכילת המנה העיקרית כלל האורחים מעוניינים לאכול קינוח בעמדת הקינוחים. מתוך המסלול הראשי יש ניסון לתפוס מקום בעמדת הקינוחים. במידה ואורה הצליח לתפוס מקום, הוא נכנס לתור לעמדה וועובר למסלול קינוחים. לאחר קבלת הקינוח האורה מרוצה ולא עוזב את האולם. במידה ואורה לא הצליח לתפוס מקום בגלל הגבלת האולם (עד 15 אנשים), הוא הולך לתת מסלול של ריקודים וחוזר ממנו בחזרה למסלול הראשי. לאחר הריקוד הוא מנסה לתפוסשוב, פעללה אשר קורית שוב 3 פעמים. במידה ולאחר הפעם השלישייה לא עלה בידו הדבר, הוא לא מנסה שוב וועוזב את האולם. ([נספח 1.5](#))

כל שאר המסלולים שעוברות היישוות עם הסברים ובליוי קטעי קוד מופיעים [בנספח 1.6](#).

5.2. הנחות

- יחידת זמן הסימולציה שנקבעה היא דקה (1 יחידת זמן = 1 דקה).
- הסימולציה מתחלילה בזמן 0, משך החתונה הינו 6 שעות (מתוכם אורחים מגיעים עד 4 שעות מתחילה האירוע).
- ישנו 8 סו-שפים בעמדת הקינוחים המסוגלים לספק קינוח. כדי שנוכל למדל העדפה מבחינת רמת שבוע הוחלט כי ישנו 8 אורחים שיכולים לאכול בעמדה עצמה ועוד 7 העומדים בתור (סה"כ 15 אורחים במתחם). העדיפות ממומשת כאשר אורחים בעלי עדיפות גבוהה יותר עוקפים רק בתור את אלו עם העדיפויות הנמוכות.
- עם תחילת הסימולציה, כמות האורחים שעולים על ההסעות ידועה (ההסעה יצאה לאירוע).
- אורחים מאוכזבים אשר לא הצלחו לקבל קינוח בפעם השלישייה עוזבים את האולם. לעומת זאת אורחים שמצליחים לאכול קינוח נשארים עד סוף החתונה ולא עוזבים. אורחים שלא הספיקו לקחת קינוח במהלך החתונה ייחסבו גם הם כמאוכזבים.
- אורח שהגיע לדוכן יכול לא עוזב אותו עד שייקבל מנה, או עד שהחופה תתחל.
- במידה וקיים בחירה בין שתי עמדות עם אותן שירות המתקבל, האורחים יילכו לעמדה בעלת התור הקצר ביותר.
- בחלוקת הוספת הפירות, העמדה נפתחת לדקה אחת וכל/OR האורחים שmagיעים לאולם יכולים לקחת מהפירות (אלו שרוקדים יסימו לרוקוד ורק לאחר מכן אם העמدة תישאר פתוחה יבואו לקחת ממנה).

6. ניתוח מצב קיימם והצעת חלופות לשיפור מצב קיים

6.1. בחירת מדדים לניתוח

ראשית, ערכנו בדיקה עבור נכונות המודל אשר נמצא תקין ([נספח 2](#)). לאחר מכן, בהתאם לאופי הארגון, מטרותיו ומטרות הפרויקט, הגדרנו את המדדים הבאים:

1. זמן המתנה ממוצע בעמדות קבלת הפנים (עמדות האוכל בלבד) – נרצה למזער מدد זה. המטרה המרכזית של הפרויקט הינה לשפר את איכות השירות. ככל שערך מدد זה יהיה נמוך יותר (אורהים ימתינו פחות בתורים), כך השירות באולם האירועים יקבע איכותו ויעיל יותר. חישוב המدد מתבצע ע"י חישוב סכום זמני המתנה הממוצעים בכל עמדה לחלק במספר העמדות הקיימות. המدد חושב באמצעות פונקציה מתאימה ([נוף 4.1](#)).

2. אחוז האורחים שיצאו מאוכזבים מהאירוע – נרצה למזער מدد זה. לקוחות שניים לחת קינוח ולא הצליחו או לקוחות שלא הספיקו לחת קינוח יצאו מאוכזבים לגבי האירוע כולו. لكن ראיינו חשיבות במצבם מספר המאוכזבים באירוע. אחוז האורחים שיצאו מאוכזבים מהאירוע יחושב ע"י מספר האורחים שלא הצליחו לחת קינוח לחלק במספר האורחים באירוע כולו. המدد חושב באמצעות פונקציה מתאימה ([נוף 4.2](#)).

3. אחוז האורחים שהגיעו לחופה בזמן – נרצה למסס מدد זה. בשל תורים גדולים בעמדות קבלת הפנים יתכן שאורחים יאחו לחופה. זוג שמתחנן ירצה שכמה שייתר אורחים ינכחו בחופה שלו, כמו כן, האורחים ירצו להיות נוכחים בחופת הזוג ולכך מKeySpec מدد זה ישפר את חווית הזוג המתחנן וכן את חווית האורחים. המدد יחושב ע"י מספר האורחים שהגיעו לחופה לפני בזמן חלק מספר האורחים הכלל. המدد חושב באמצעות פונקציה מתאימה ([נוף 4.3](#)).

6.2. סוג המערכת

המערכת שלנו היא מסווג מערכת מסתיימת – System Terminating, כיוון שבמנן את פעילותם אולם האירועים לאורך יומם העבודה אחד. על כן אין צורך בזמן חיים. ייחדות הזמן שבמודל הוגדרו להיות בדקות. משך זמן של הרצה בודדת הוא 6 שעות, כלומר 360 דקות.

מספר הריצות הראשוני שבחרנו בצורה שירוטית הינו $n_0 = 20$.

לצורך קביעת מספר הריצות המינימלי הדרוש עבור כל מערכת בודדת, בדקנו האם מתקיים איילוץ

$$\frac{\delta(n,a)}{\bar{x}} \leq \frac{\gamma}{1+\gamma}.$$

בחרנו את הדיק היחסי להיות $\gamma = 0.1$, עברו החסם המתקבל הינו $0.090909 = \frac{\gamma}{1+\gamma}$.

בחרנו את רמת המובהקות הכוללת להיות $\alpha_{total} = 0.09$, וחילקו במספר המדים. רמת

$$\text{المובהקות עבור כל ממד היא } \alpha_i = \frac{\alpha_{total}}{3} = \frac{0.09}{3} = 0.03.$$

את ערכי הממוצע וסטיית התקן עבור כל המדים בכל חלופה חישבנו לפי 20 הריצות. ([נוף 5.1](#))

$$\frac{\delta}{\bar{x}} = \frac{t_{(20-1),(1-\frac{0.03}{2})^{\frac{0.03}{2}} * \sqrt{20}}}{\text{ממוצע}}$$

להלן תוצאות הדיק היחסי לכל חלופה וממד שהתקבלו, על פי הנוסחה

([נוף 5.2](#))

מדד אחות האורחים שהגיעו לחופה בזמן	מדד אחות האורחים שיצאו מאוכזבים מהאיירע	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	
0.031	0.0072	0.0753	מצב קי"ם
0.025	0.0327	0.0864	חלופה 1
0.029	0.0375	0.1022	חלופה 2

ניתן לראות כי חלופה 2 לא עומדת בדיק היחסי, אך נצטרך להוסיף ריצות נוספות. נשערך את המספר הריצות הנדרש על ידי הדיק הגס המחשב בעזרת הנוסחה:

$$n = n_0 * \left(\frac{\delta_0}{X * \frac{\gamma}{1 + \gamma}} \right)^2$$

עבור מצב קי"ם התקבל $n = 14$, עבור חלופה 1 התקבל $n = 19$ ועבור חלופה 2 התקבל $n = 26$. לכן נבחר את הגבהה מ בין השלושה ונאמר כי דרישות בסך הכל 26 ריצות. החלטנו להריץ 30 ריצות מפני שרצינו להבטיח עמידה בדיק היחסי וכן הדבר מתאפשר מבחינת שיקולי זמן. ([סעיף 5.2](#))

לאחר הריצת הריצות הנוספות שנדרשו, נמצא כי מספר הריצות החדש שבצענו מספיק עבור רמת המובاهקות ורמת הדיק היחסי שהגדכנו קודם לכן. ([סעיף 5.3](#))

6.3. ניתוח מצב קי"ם ביחס למדדים ולמטרות

להלן תוצאות המדדים של 30 ריצות במצב הקי"ם: ([נספח 5.4](#))

מדד אחות האורחים שהגיעו לחופה בזמן	מדד אחות האורחים שיצאו מאוכזבים מהאיירע	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	
0.2761	0.265	21.3751	ממוצע
0.0132	0.0106	2.7977	סטיית תקן

ניתוח תוצאות המדדים- המצב הקי"ם:

מדד זמן המתנה ממוצע בעמדות קבלת הפנים – ניתן לראות כי במצב הקי"ם זמן המתנה הממוצע של האורחים בעמדות קבלת הפנים הינו 21.3751 דקות. מדובר בנתון יחסית גובה. מצב זה מעיד שעמדות קבלת הפנים עומסות, יכול להיות בשל חוסר בכוח אדם או בכמות העמדות שנעמדו לשרת את כלל האורחים.

מדד אחות האורחים שיצאו מאוכזבים מהאיירע – ניתן לראות כי במצב הקי"ם אחוז האורחים שיצאו מאוכזבים מהאיירע עומד על 26.5%. מעל רביע מהאורחים יוצאים מאוכזבים ונצטרך לשפר נתון זה. מצב זה יכול להעיד על הרבה אורחים שלא מספיקים להגיא לאזרור הקינוחים בשל עומס בעמדות הקודמות או שאינם מצלחים לקחת קינוח בשל עומס בעמדת הקינוחים עצמה.

מדד אחוז האורחים שהגיעו לחופה בזמן – ניתן לראות כי במצב הנוכחי אחוז האורחים שנכנסו לחופה עומד על 27.61% . זה נתון נמוך ונרצה שיתשתפר. הדבר יכול להעיד על עומסים גבוהים בעמדות קבלת הפנים שבגללן האורחים אינם מספיקים לחופה.

6.4. בחירת החלופות

1. חלופה ראשונה

עלות	תיאור	шиיפור
20,000 ₪	הוסף מלצר לאחת מעמדות הטורטיות ומלצר לעמדת הטורטיות	הוסף 2 מלצרים
30,000 ₪	הוסף עמדת פירות שתפתח 30 דקות לאחר סיום החופה למשך דקה אחת	הוסף עמדת פירות
50,000 ₪		סה"כ

כיוון שהמדד הראשון עוסק בזמן הרמתנה הממוצע בעמדות קבלת הפנים בחרנו להוסיף 2 מלצרים ל2 מעמדות קבלת הפנים בהנחה שלא ימצאו את המدد. כמו כן, במידה וזמן הרמתנה הממוצע בעמדות ישתרף, האורחים יסיעמו עם עמדות קבלת הפנים מהר יותר וכן יותר אורחים יגיעו לחופה בזמן, מה שיעזר בשיפור המدد השלישי – מקסום כמות האורחים הנוכחים לחופה. המدد השני עוסק במקסום האורחים הלא מאוצבים ולכן בחרנו להוסיף עמדת פירות, כך שע"י חלוקת הפירות האורחים לא יצטרכו קינוח וייהו מרוצים מהארוע.

להלן תוצאות המדדים לאחרนำไปש השיפורים המוצעים בחולפה 1 : ([גופח 5.4](#))

מדד אחוז האורחים שהגיעו לחופה בזמן	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	מדד זמן המתנה ממוצע בעמדות קבלת הפנים
0.398	0.2369	13.967	ממוצע
0.0228	0.0175	3.212	סטיית תקן

נתוחות תוצאות המדדים – חלופה 1:

מדד זמן המתנה ממוצע בעמדות קבלת הפנים – ניתן לראות שבחלופה זו זמן הרמתנה הממוצע בעמדות קבלת הפנים הינו 13.967 דקות, מדובר בשיפור של 7.4 דקות ביחס לזמן הרמתנה הממוצע. ניתן לראות כי ערך המدد ירד בצורה מובהקת בזכות חלופה זו אם זאת, כאמור, זמן הרמתנה הממוצע בחולפה זאת נמור יותר.

מדד אחוז האורחים שיצאו מאוצרם מהארוע – ניתן לראות שבחלופה זו אחוז האורחים שיצאו מאוצרם עומד על 23.69% . מדובר בשיפור של 2.81% בערך המدد. ניתן לראות כי ערך המدد ירד בצורה מובהקת בזכות חלופה זו, כאמור, אחוז האורחים המאוצרם נמור יותר בחולפה זו.

מדד אחוז האורחים שהגיעו לחופה בזמן – ניתן לראות שבחולופה זו אחוז האורחים שנכחו בחופה עומד על 39.8%. מדובר בשיפור של 12.19% בערך המدد, כלומר, אחוז הנוכחים בחופה בחולופה זו גבוה יותר מהמצב הנוכחי בזורה די משמעותית.

2. חלופה שנייה

שיעור	תיאור	עלות
הוסףת 2 מלצרים	הוסףת מלצר לאחת מעמדות הסושי ומלצר לעמדת הצמחונית	20,000 ₪
הוסףת עמדת טורטיות נוספת	הוסףת עמדת טורטיות נוספת בעמדות קבלת הפנים	25,000 ₪
סה"כ		45,000 ₪

בדומה לשיבוט שפורטו בחולופה הראשונה, גם בחולופה זאת שמננו דגש על צמצום זמן המתנה הממוצע בעמדות קבלת הפנים וכן על מיקסום מספר האורחים הנוכחיים בחופה. על כן, הוסףנו 2 מלצרים ועמדת טורטיות נוספת בעמדות קבלת הפנים. בחרנו להוסיף מלצר לאחת מעמדות הסושי כיוון שלעמدهה זאת מספר המלצרים הנמור ביותר.

להלן תוצאות המדדים לאחר מימוש השיפורים המוצעים בחולופה 2 : ([טוף 5.4](#))

מדד אחוז האורחים שהגיעו לחופה בזמן	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	מדד אחוז האורחים שיצאו מאoczבים מהארוע	מדד זמן המתנה ממוצע בעמדות קבלת הפנים	
0.45	14.384	0.24	ממוצע	
0.0271	2.847	0.0175	סטיית תקן	

ניתנו תוצאות המדדים – חלופה 2:

מדד זמן המתנה ממוצע בעמדות קבלת הפנים – ניתן לראות שבחולופה זו זמן המתנה הממוצע בעמדות קבלת הפנים הינו 14.384 דקות, מדובר בשיפור של 7 דקות בזמן המתנה הממוצע. ניתן לראות כי ערך המدد ירד בצורה מוגבלת בזכות חולופה זו אם זאת. שיפור המدد אICONTI אך נמוך כמעט מהחולופה הקודמת.

מדד אחוז האורחים שיצאו מאoczבים מהארוע – ניתן לראות שבחולופה זו אחוז האורחים שיצאו מאoczבים עומד על 2.4%. מדובר בשיפור של 2.5% בערך המدد. חולופה זו שיפורה את ערך המدد, כלומר, פחות אורחים יצאו מאoczבים מהארוע בחולופה זו.

מדד אחוז האורחים שהגיעו לחופה בזמן – ניתן לראות שבחולופה זו אחוז האורחים שנכחו בחופה עומד על 45%. מדובר בשיפור של 17.39% בערך המدد. כלומר, אחוז הנוכחים בחופה בחולופה זו גבוה משמעותית מהמצב הנוכחי ומתקרב לכמעט מחצית מהאורחים.

6.5. השוואת סטטיסטי בין מצב קיימלופות מוצעות

לאחר בוחנת המצב הקיים ושתי החלופות על פי שלושת הממדים, נחליט איזו חלופה היא אופטימלית. לשם כך, ביצענו מבחן t מזווג. ([נספח 5.5](#))

מדד	השוואה	רוח סמן	תוצאות המבחן	חלופה מועדף עבור כל הממד
זמן המתנה מוצע בתור לעמדות קבלת הפנים	מצב קיים – חלופה 1	(6.126, 8.689)	חלופה 1	חלופה 1 / חלופה 2
	מצב קיים – חלופה 2	(5.619, 8.361)	חלופה 2	
	אדישות	(-1.960, 1.126)	חלופה 1 – חלופה 2	
אחוז האורחים שיצאו מאוכזבים מהארוע	מצב קיים – חלופה 1	(0.020, 0.035)	חלופה 1	חלופה 1 / חלופה 2
	מצב קיים – חלופה 2	(0.017, 0.032)	חלופה 2	
	אדישות	(-0.007, 0.001)	חלופה 1 – חלופה 2	
אחוז האורחים שהגיעו לחופה בזמן	מצב קיים – חלופה 1	(-0.129, -0.114)	חלופה 1	חלופה 2
	מצב קיים – חלופה 2	(-0.182, -0.165)	חלופה 2	
	חלופה 2	(-0.06, -0.04)	חלופה 1 – חלופה 2	

עבור הממדים **זמן המתנה** **מוצע בתור לקבלת הפנים**, **אחוז האורחים שיצאו מאוכזבים מהארוע** לא ניתן להכריע בצורה מובהקת מי משתי החלופות טוביה יותר. עם זאת, בשני הממדים החלופות טובות מהמצב הקיים. עבור הממד **אחוז האורחים שהגיעו לחופה בזמן** חלופה 2 היא העדיפה וכן ניתן לומר שמדובר עליה.

7. מסקנות

מטרת הפרויקט היא למצוא דרכי ליעול התנהלות אולם האירועים בתקציב של עד 50,000 ₪. הייעול נדרש לתת חוות יותר נעימה לאורחים (SHIPOR חוות תבטוא בשיפור ערכי הממדים). בחנו שתי חלופות אפשריות:

1. **תוספת 2 מלצרים + הוספת עמדת פירוט בעלות של 50,000 ₪.**
2. **תוספת 2 מלצרים + הוספת עמדת טורטיות נוספת בעלות של 45,000 ₪.**

התוצאות שהתקבלו בעת בוחנת המצב הקיים והחלופות באמצעות סימולציה. באופן כללי 2 החלופות שבחנו עדיפות על המצב הקיים ובשני הממדים הראשונים (זמן המתנה ממוצע בתור לקבלת הפנים ואחוז האורחים שיצאו מאוכזבים מהארוע) ישנה אדישות בין חלופה 1 לחלופה 2, אך במדד השלישי (אחוז האורחים שהגיעו לחופה בזמן) חלופה 2 עדיפה. על כן, כיוון שחלופה 2 עשויה להביא לשיפור כלל הממדים הימנו ממליצים עליה.

לטענו מטרת הפרויקט הושגה- שיפרנו בצורה ניכרת את שלושת הממדים שבחנו. נסיק מכאן שהמספרה יכולה להסתמך על המודל שהצענו והוא עתיד לשפר את המטרת הרצiosa.

נספח 1 – מודל הסימולציה

נספח 1.1 – מסלול ראשי

```

mainPath<- trajectory("mainPath")%>%
  set_attribute("type", function () get_type()%>%
    join(reception)%>%
    # log_(in finish the reception)%>%
    set_attribute("disappointed", 1)%% # disappointed
    clone(function() get_attribute(weddingSimulation, "type"))%>%
    set_attribute(keys=c("food1","food2","food3","food4","food5"), values = function() favouriteFoods())%>%
    set_attribute("counter", 1)%>%
    branch(option = function() get_food(), continue = c(TRUE,TRUE,TRUE,TRUE,TRUE),
           tortillasPath,vegsPath,sushiPath,bunsPath,focacciaPath)%>% #split to food stands
    set_attribute("counter", 1, mod = "+")%>%
    rollback(times = 4, amount = 2)%>%
    # log_(finish dishes)%>%
    synchronize(wait = TRUE, mon_all = TRUE)%>%
    # log_(finish all dishes)%>%
    seize("huppa",1)%% # everyone waiting until the end of the huppa
    release("huppa",1)%>% # huppa end
    # log_(huppa is over)%>%
    branch (option = function() getType(), continue = c(TRUE),
            internalBarPath)%>% # sending couples and singles to the internal bar
    timeout(rexp(1,1/3.5))%>% # time to search for table
    # log_(arrived to table)%>%
    timeout(function() groupEatingTime())%>%
    # log_(finish to eat)%>%
    set_attribute("satiety_level", function() groupsatiety(get_attribute(weddingSimulation, "type")))%>%
    set_prioritization(values = function() returnsatiety())%>%
    # log_(arrived to dessert queue)%>%
    seize("desserts",1, continue = c(FALSE, TRUE), post.seize = dessertPath,
          reject = dancePath)%>% #if catch dessert go to eat
    rollback(times = 2, amount = 1)%>%
    leave(1)
  
```

נספח 1.2 – מסלול טריגר reset

```

resetPath <- trajectory("resetPath")%>%
  set_global(keys = "busTime", value = function() runif(1,60,75))%>%
  set_global(keys = "singlesQunty", value = function() floor(runif(1,100,151)))%>%
  set_global(keys = "weddingTime", value = function() runif(1,140,160))%>%
  set_global(keys = "weddingLength", value = function() runif(1,20,35))%>%
  set_global(keys = "huppaEnd", value = function() get_global(weddingSimulation,key="weddingTime") + get_global(weddingSimulation, key="weddingLength"))%>%
  set_global(keys = "barClose", value = function() get_global(weddingSimulation,key="weddingTime") - 10)%>%
  set_queue_size("reception1",Inf)%>%set_capacity("reception1", 2)%>%
  set_queue_size("reception2",Inf)%>%set_capacity("reception2", 2)%>%
  set_queue_size("focaccias",Inf)%>%set_capacity("focaccias", 3)%>%
  set_queue_size("buns",Inf)%>%set_capacity("buns", 3)%>%
  set_queue_size("sushi1",Inf)%>%set_capacity("sushi1", 1)%>%
  set_queue_size("sushi2",Inf)%>%set_capacity("sushi2", 2)%>%
  set_queue_size("vegs",Inf)%>%set_capacity("vegs", 4)%>%
  set_queue_size("tortillas",Inf)%>%set_capacity("tortillas", 4)%>%
  set_queue_size("externalBar",Inf)%>%set_capacity("externalBar", 3)%>%
  set_queue_size("internalBar1",Inf)%>%set_capacity("internalBar1", 5)%>%
  set_queue_size("internalBar2",Inf)%>%set_capacity("internalBar2", 7)%>%
  set_queue_size("desserts",7)%>%set_capacity("desserts", 8)%>%
  set_queue_size("parking",Inf)%>%set_capacity("parking", Inf)%>%
  set_queue_size("huppa",Inf)%>%set_capacity("huppa", 0)%>%
  timeout_from_global("barClose")%>%
  set_queue_size("externalBar",0)%>%
  timeout(10)%>%
  set_queue_size("focaccias",0)%>%
  set_queue_size("buns",0)%>%
  set_queue_size("sushi1",0)%>%
  set_queue_size("sushi2",0)%>%
  set_queue_size("vegs",0)%>%
  set_queue_size("tortillas",0)%>%
  timeout_from_global("weddingLength")%>%
  set_capacity("huppa", Inf)
  
```

נספח 1.3 – דוגמא ל أحد מסלולי דוכן האוכל (Ban)

```

bunsPath<-trajectory("bunsPath")%>%
  #log_(in in bunPath)%>%
  addService("buns", function() trimmedNorm(1.5, 0.7))%>%
  timeout(rexp(1,0.8))%>% #time for eating the food
  branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),
                                             chanceForBarByType()),c(0,1)), continue = c(TRUE) ,externalBar)%>%
  timeout(0)
  
```

נספח 1.4 – מסלול בר חיצוני

```
internalBarPath<-trajectory("internalBarPath")%>%  
  simmer::select(resources=c("internalBar1","internalBar2"), policy=c("shortest-queue"))%>%  
  #log_("im in internalBar")%>%  
  seize_selected(amount = 1) %>%  
  timeout(rexp(1,2.4)) %>%  
  release_selected(amount = 1)
```

נספח 1.5 – מסלול קינוחים וריקודים

```
seize("desserts",1, continue = c(FALSE, TRUE), post.seize = dessertPath,  
      reject = dancePath)%>% #if catch dessert go to eat  
rollback(times = 2, amount = 1)%>%  
leave(1)  
  
dessertPath<- trajectory("dessertPath")%>%  
  log_("I catch the dessert")%>%  
  timeout(function() runif(1,2.5,4))%>% #people eat in dessert area  
release("desserts",1)%>%  
set_attribute("disappointed", 0) # not disappointed
```

נספח 1.6 – תיאור כל המסלולים שהישויות עוברות כשהן מגיעות לאירוע

דבר ראשון, היחסיות נוצרות על פי ההתפלגיות מקובץ ההוראות.

- ישנן שתי יישויות המדומות טריגרים: "trigger" – מפעיל את הגעת היחידים מהאוטובוס, "reset"
- ישות אשר מטרתה לאפס את כל המשאים (פונקציית reset לא איפosa בזרה טובה) ובנוסף מטרת הישות להגדיר מחדש משתנה גלובליים מתאימים בכל תהליך ריצה חדש, לסגור את הבר החיצוני 10 דקות לפני תחילת החופה ולשחרר את משאב החופה לאחר סיוםה.

```
##----- 5. All Generators, ALWAYS LAST. -----  
weddingSimulation%>%  
  add_generator("reset", resetPath, distribution = at(0))%>%  
  add_generator("couple", parkingPath, distribution = to(4^60, function () rexp(1,lambda_couple)), mon=2)%>%  
  add_generator("family", parkingPath, distribution = to(4^60, function () rexp(1,lambda_family)), mon=2)%>%  
  add_generator("single", mainPath, when_activated=function() get_global(weddingSimulation, "singlesQuantity")), mon=2)%>%  
  add_generator("trigger", trigger, at(0))
```

התפלגות המשפחות והזוגות נקבעו על ידי קובץ האקסל שקיבלנו:

```
> #read data of family and couple  
> data <- read.csv(file.choose(), header = T)  
> diff_couples<-data$couples_diff/60 #converting to minutes from seconds  
> #create fitdist for some distribution  
> couple_Fit_exp <- fitdist(diff_couples,"exp")  
> print(paste("Best distribution for couples is EXP with lambda:",round(couple_Fit_exp$estimate[1], 5)))  
[1] "Best distribution for couples is EXP with lambda: 0.55883"  
> diff_families<-data$families_diff/60  
> family_Fit_exp <- fitdist(diff_families,"exp")  
> print(paste("Best distribution for families is EXP with lambda:",round(family_Fit_exp$estimate[1], 5)))  
[1] "Best distribution for families is EXP with lambda: 0.81193"
```

היחידים מתחילהם להיווצר כאשר trigger מופעל:

```
trigger<- trajectory("trigger") %>%
  timeout_from_global("busTime")%>%
  activate("single")
```

הזוגות והמשפחות מחפשות מקום לחניה ולאחר מכן מגיעות למסלול הראשי:

```
parkingPath <- trajectory("parkingPath" ) %>%
  #log_( "im in parking")%>%
  addService("parking", function() rtriangle(1,3, 5, 4))%>%
  join(mainPath)
```

כאשר המוזמנים מגיעים למסלול הראשי (1), מוגדרת להם תוכנת ה-Type ע"פ פונקציית get_type, שמחלקת את המשפחות לסוג [3,4,5] לפי ההסתברויות שנתקבשנו, זוגות ל-2 ויחידים ל-1. כל היסיות עוברות למסלול הקבלה לקבלת שולחן, כאשר הבחירה באיזה קבלה לכת מתבצעת לפי התווך הקצר ביותר שהן רואות ("shortest-queue"). 7 מהמוזמנים מבינים שהגיעו לאולם הלא נכון ועזבים. אלו שנשארים מקבלים שולחן לפי ההתפלגות הנדרשת.

```
mainPath<- trajectory("mainPath")%>%
  set_attribute("type", function () get_type()%>%
    join(reception)%>%
    #log_( "Im finish the reception")%>%

reception<- trajectory("reception")%>%
  simmer::select(resources=c("reception1","reception2"), policy=c("shortest-queue"))%>%
  #log_( "im in reception")%>%
  seize_selected(amount = 1) %>%
  timeout(trimmedNorm(0.5, 0.2)) %>%
  release_selected(amount = 1)%>%
  leave(0.07)%>%
  set_attribute("table", function () floor(runif(1,1,96)))

# return type of guest by number of guests
get_type <- function(){
  name<- get_name(weddingSimulation)
  x <- 0
  if (substr(name,0,1) == "f"){ # family
    x <- rdiscrete(1,c(0.33,0.4,0.27), c(3,4,5))
  }
  else if (substr(name,0,1) == "c") {x <- 2} #couple
  else {x <- 1} #single
  return (c(x))
}
```

לאחר קבלת השולחן, האורחים מתפצלים ליחידים וניגשים לעמדות האוכל השונות, כל אחד לפי הסדר שנקבע לו על ידי פונקציית favouriteFoods(). מידלונו את הפעולה החזורת של לקיחת אוכל מדוכן ולאחריו בהסתברות מסוימת הליכה לביר בעזרת() Rollback. כל מסלול של דוכן מוגדר כתת מסלול אליו מגיע האורח מתוך המסלול הראשי. לאחר המעבר על כל תתי המסלולים (יחידים עוברים רק ב-3 העמדות הרלוונטיות אליהם), מתבצע הסyncronon מחדש של המשפחות והזוגות, כאשר הם יכולים להתקדם לחופה רק כאשר כל בני המשפחה סיימו לאכול (wait = TRUE).

```

#log_("In finish the reception")%>%
clone(function() get_attribute(weddingSimulation, "type"))%>%
set_attribute(keys=c("food1","food2","food3","food4","food5"), values = function() favouriteFoods())%>%
set_attribute("counter", 1)%>%
branch(option = function() get_food(), continue = c(TRUE,TRUE,TRUE,TRUE,TRUE),
       tortillasPath,vegsPath,sushiPath,bunsPath,focacciaPath)%>% #split to food stands
set_attribute("counter", 1, mod = "+")%>%
rollback(times = 4, amount = 2)%>%
# log_("finish dishes")%>%
synchronize(wait = TRUE, mon_all = TRUE)%>%

```

דוגמא לאחד מדוכני האוכל- סושי. עבור מסלול זה, אורח שורצה סושי יגש לתוך הזמן הקצר ביותר:

```

sushiPath<-trajectory("sushiPath")%>%
simmer::select(resources=c("sushi1","sushi2"), policy=c("shortest-queue-available"))%>%
#log_("in in sushiPath")%>%
seize_selected(amount = 1) %>%
timeout(trimmedNorm(1.5, 0.7)) %>%
release_selected(amount = 1)%>%
timeout(rexp(1,0.8))%>%
branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()),
                                         |c(0,1)), continue = c(TRUE),externalBar)%>%
timeout(0)

```

מסלול הבר החיצוני:

```

externalBar <- trajectory("externalBar")%>%
#log_("In drinking at external bar")%>%
addService("externalBar", function () rexp(1,2))

```

לאחר סיום כל הדוכנים האורחים ממתינים לתחילת החופה, ע"י תפיסת המשאבות חופה. 10 דקות לפני תחילתה ישות "reset" מעדכנת את אורך התווך ל-0 ועם תחילת החופה הרישוט מעדכנת את אורך התווכים של דוכני האוכל ל-0 ואז האורחים מתקדים אל החופה. התוצאה של משאב החופה נקבעת גם כן ע"י ישות "reset".

```

timeout_from_global("weddingLength")%>%
set_capacity("huppa", Inf)

```

שהחופה נגמרה, כל האורחים משתחררים ממנה בביטחון אל המשך החתונה, כאשר הזוגות והיחידים הולכים לביר הפנימי.

```

seize("huppa",1)%>% # everyone waiting until the end of the huppa
release("huppa",1)%>% # huppa end
#log_("huppa is over")%>%
branch (option = function() getType(), continue = c(TRUE),
         |internalBarPath)%>% # sending couples and singles to the internal bar
timeout(rexp(1,1/3.5))%>% # time to search for table
# log_("arrived to table")%>%

```

לאחר מזיאת השולחן, כל האורחים אוכלים. ניתן להתקדם מבחןכם רק לשלב הבא רק כאשר כל בני המשפחה סיימו לאכול, אך השתמשנו בפונקציית groupEatingTime() שמצוצת מה זמן האכילה הארוך ביותר לאחד מבני המשפחה, ואת זמן זה תמתין כל המשפחה.

```

timeout(function() groupEatingTime())%>%
# log_("finish to eat")%>%

```

תיאור הפונקציה:

```
groupEatingTime<-function(){ #The function samples the eating times of all
  #members of the family/couple/individual and returns
  #the highest eating time among all members of the group
  familyMembers<- get_attribute(weddingSimulation,"type") # number of guest in the group
  t<- rdiscrete(familyMembers,c(0.7,0.3),c(1,2)) # with a probability of 70 percent -
  #a meat dish and with a probability of
  #30 percent - a vegetarian dish
  x<-c(0,0,0,0,0)
  n=1
  for(i in t){
    if(i==1) {x[n]<-trimmedNorm(12,3)} #meat
    else {x[n]<-trimmedNorm(7,4)} #vegetarian
    n=n+1
  }
  maxTime <-max(x) #the highest from group's member
  return(maxTime)
}
```

לאחר סיום הארוחה אנו מגדרים את אורחיהם את רמת השובע שלהם, וע"פ רמת השובע מוגדרת העדיפות לקינוחים כאשר כל מי שיש לו רמת שבוע > 4.5 מקבל עדיפות 0 וכי שלא מקבל עדיפות 1.

```
set_attribute("satiety_level", function() groupSatiety(get_attribute(weddingSimulation,"type")))%>%
  set_prioritization(values = function() returnSatiety())%>%
  # log_("arrived to dessert queue")%>%
  seize("desserts",1, continue = c(FALSE, TRUE), post.seize = dessertPath,
        reject = dancePath)%>% #if catch dessert go to eat
  rollback(times = 2, amount = 1)%>%
  leave(1)
```

לאחר מכון האורחים מנוטים ללקחת קינוח, אלו שנדרחים, בגלל שהוזיאו אותם מהתו מבחןת עדיפות או כי אין מקום במתחם (עד 15 אנשים), הולכים למסלול הריקודים. במידה ולא הצליחו לתפוף 3 פעמים את עמדת הקינוחים הם עוזבים את האולם.

אלו שתפסו את העמדה, אוכלים את הקינוח ומואושרים וכך נשארים באולם עד סיום החתונה.

```
dessertPath<- trajectory("dessertPath")%>%
  log_("I catch the dessert")%>%
  timeout(function() runif(1,2.5,4))%>% #people eat in dessert area
  release("desserts",1)%>%
  set_attribute("disappointed", 0) # not disappointed
```

נספח 2 – בדיקת נכונות המודל

עבור כל אחד מסוגי הישויות (משפחות, זוגות, יחידים) ביצענו בדיקה שהישות עוברת את המסלול בצורה נכונה. עבור כל בדיקה נטרלנו את שאר היחסיות, יצרנו ישות אחת מסווג ספציפי ועקבנו אחר מסלולה.

בדיקה עבור אורח מסווג 'יחיד':

ראשית, נטרלנו את כל היחסות ויצורנו ישות 'יחיד' אחת:

```
##----- 5. All Generators, ALWAYS LAST. -----
weddingSimulation%>%
  add_generator("single", mainPath, whenActivated(1), mon=2) %>%
  # add_generator("couple", parkingPath, distribution = to(4*60, function () rexp(1,lambda_couple)), mon=2)%>%
  # add_generator("family", parkingPath, distribution = to(4*60, function () rexp(1,lambda_family)), mon=2)%>%
  # add_generator("single", mainPath, whenActivated(singlesQuantity), mon=2) %>%
  add_generator("trigger", trigger, at(busTime))%>%
  add_generator("barTrigger", closeBar, at(weddingTime - 10))%>%
  add_generator("dishesTrigger", closeDishes, at(weddingTime))
```

ננתן את הפלט עבור ישות זו: (מס' רשומה המסומן בכוכבית משמעו שמדובר ברשומה מהטבלה התחתונה)

	name	start_time	end_time	activity_time	resource
1	single0	71.36224	72.28771	0.92546561	reception1
2	single0	72.28771	72.93376	0.64605299	tortillas
3	single0	73.01133	73.16345	0.15211686	externalBar
4	single0	73.16345	75.25712	2.09367778	sushi1
5	single0	75.90430	76.06977	0.16547396	externalBar
6	single0	76.06977	78.17588	2.10610723	vegs
7	single0	81.72023	82.18817	0.46794009	externalBar
8	single0	82.18817	185.07947	0.00000000	huppa
9	single0	185.07947	185.11452	0.03504747	internalBar2
10	single0	196.63998	200.21943	3.57944474	desserts

(1) 'יחיד' מגיע לקבלה מספר 1

(*) לאחר הסיום בקבלה, מקבל מס' שולחן (46)

(2) מגיע לעמדת הטורטיות

(3) מגיע לביר החיצוני

(4-7) המשך התנהלות בעמדות קבלת הפנים

(8) הגעה לחופה

(9) הגעה לביר הפנימי השני

(*) קבלת רמת שבוע לאחר שאכל (רמת שבוע 2.68)

(10) אכל קינוח

(*) האורח לא מאוכזב וממתין לסיום האירוע

הישות עוברת את המסלול בצורה נכונה.

בדיקה עבור אורח מסוג 'זוג':

ראשית, נטרלנו את כל הישויות ויצרנו ישות 'זוג' אחת:

```
##----- 5. All Generators, ALWAYS LAST. -----#
weddingSimulation<-%>
  add_generator("couple", parkingPath, distribution = at(5), mon=2)%>%
  # add_generator("couple", parkingPath, distribution = to(4*60, function () rexp(1,lambda_couple)), mon=2)%>%
  # add_generator("family", parkingPath, distribution = to(4*60, function () rexp(1,lambda_family)), mon=2)%>%
  # add_generator("single", mainPath, when_activated(singlesQuantity), mon=2) %>%
  # add_generator("trigger", trigger, at(busTime))%>%
  add_generator("barTrigger", closeBar, at(weddingTime - 10))%>%
  add_generator("dishesTrigger", closeDishes, at(weddingTime))
```

ננתן את הפלט עבור ישות זו: (מס' רשותה המופיע בכוכבית משמעו שמדובר ברשותה מהטבלה התחתונה)

(1) 'זוג' מגיע לאזרור החניה

(2) הגעה לקבלת מס' 1

(*)2 קבלת מס' שולחן (שולחן 28)

הזוג מתפצל#

(3) אחד מבני הזוג מגיע לעמדת הטורטיות

(4) אחד מבני הזוג מגיע לעמדת הפוקצ'ות

(5-20) המשך התנהלות בעמדות קבלת הפנים

הזוג מתאחד#

(18) הזוג מגיע לחופה

(19) הזוג מגיע לבר הפנימי השני

(*)25) הזוג מקבל רמת שבוע לאחר שאכל (רמת שבוע 3.29)

(20) הזוג אכל קינוח

(*)26) הזוג לא מאוכזב וממתין עד שהAIRU יגמר

הישות עוברת את המסלול בצורה נכונה.

	name	start_time	end_time	activity_time	resource
1	couple0	5.00000	8.95209	3.95209048	parking
2	couple0	8.95209	9.50220	0.55010958	reception1
3	couple0	9.50220	11.27344	1.77123833	tortillas
4	couple0	9.50220	12.21806	2.71585532	focaccias
5	couple0	12.58928	15.06403	2.47475002	vegs
6	couple0	15.21601	15.69765	0.48164298	externalBar
7	couple0	15.69765	17.06295	1.36529190	tortillas
8	couple0	16.10810	17.26233	1.15423605	externalBar
9	couple0	17.26233	17.77129	0.50895212	focaccias
10	couple0	18.37879	19.00210	0.62331148	externalBar
11	couple0	19.00210	20.17810	1.17600152	vegs
12	couple0	20.76924	20.98316	0.21391856	externalBar
13	couple0	21.22217	21.57702	0.35485490	externalBar
14	couple0	20.98316	21.93767	0.95450676	sushi1
15	couple0	21.57702	22.70687	1.12984405	buns
16	couple0	22.74187	22.86788	0.12600693	externalBar
17	couple0	22.86788	22.96864	0.10075904	buns
18	couple0	24.24487	24.31261	0.06774078	externalBar
19	couple0	23.98310	24.93761	0.95450676	sushi1
20	couple0	25.74181	25.82999	0.08817991	externalBar
21	couple0	25.82999	187.44111	0.00000000	huppa
22	couple0	187.44111	187.68144	0.24033158	internalBar2
23	couple0	201.11965	205.02994	3.91029124	desserts

	time	name	key	value
1	8.95209	couple0	type	2.000000
2	9.50220	couple0	table	28.000000
3	201.11965	couple0	Satiety_level	3.290196
4	205.02994	couple0	disappointed	0.000000

בדיקה עבור אורה מסוג 'משפחה':

ראשית, נטרלנו את כל הישויות ויצרנו ישות 'משפחה' אחת:

```
weddingSimulation%>%
  add_generator("family", parkingPath, distribution = at(5), mon=2)%>%
  # add_generator("couple", parkingPath, distribution = to(4*60, function () rexp(1,lambda_couple)), mon=2)%>%
  # add_generator("family", parkingPath, distribution = to(4*60, function () rexp(1,lambda_family)), mon=2)%>%
  # add_generator("single", mainPath, when_activated(singlesQuantity), mon=2) %>%
  # add_generator("trigger", trigger, at(busTime))%>%
  add_generator("barTrigger", closeBar, at(weddingTime - 10))%>%
  add_generator("dishesTrigger", closeDishes, at(weddingTime))
```

ננתן את הפלט עבור ישות זו: (מס' רשומה המסומן בכוכבית משמעו שמדובר ברשומה מהטבלה התחתונה)

#	name	start_time	end_time	activity_time	resource
1	family0	5.00000	8.95209	3.95209048	parking
2	family0	8.95209	9.50220	0.55010958	reception1
3	family0	9.50220	10.86749	1.36529190	buns
4	family0	9.50220	11.05196	1.54975984	tortillas
5	family0	9.50220	11.97695	2.47475002	focaccias
6	family0	12.14373	12.64494	0.50121878	vegs
7	family0	13.68901	13.80052	0.11150456	externalBar
8	family0	12.36780	13.94534	1.57754358	vegs
9	family0	13.80052	14.75502	0.95450676	sushi1
10	family0	14.98941	15.10665	0.11723848	externalBar
11	family0	15.10665	15.84409	0.73743547	focaccias
12	family0	14.97491	16.24504	1.27013118	buns
13	family0	15.55923	16.89739	1.33816028	focaccias
14	family0	17.52127	19.48402	1.96274771	externalBar
15	family0	18.84204	19.79655	0.95450676	sushi1
16	family0	20.60075	20.95619	0.35544047	externalBar
17	family0	19.48402	21.19133	1.70730719	vegs
18	family0	19.89534	21.44995	1.55460606	tortillas
19	family0	20.95619	21.45298	0.49678470	buns
20	family0	22.76579	23.30266	0.53687275	externalBar
21	family0	22.23540	23.43861	1.20321462	tortillas
22	family0	24.75445	25.70896	0.95450676	sushi1
23	family0	26.51316	26.54641	0.03325322	externalBar
24	family0	26.54641	187.44111	0.00000000	huppa
25	family0	202.20548	205.30364	3.09815925	desserts

(1) 'משפחה' בגודל 3 מגיעה לאזור החניה

(2) הגעה לקבלה מס' 1

(*)2 קבלת מס' שולחן (שולחן 18)

#המשפחה מתפצלת #

(3) אחד מבני המשפחה מגיע לעמדת הבאים

(4) אחד מבני המשפחה מגיע לעמדת הטורטיות

(5) אחד מבני המשפחה מגיע לעמדת הפוקאצ'ות

(6-23) המשך התנהלות בעמדות קבלת הפנים

#המשפחה מתאחדת #

(24) המשפחה מגיעה לחופה

(4.56) המשפחה מקבלת רמת שבוע לאחר שאכלו (4.56)

(25) המשפחהأكلת קינוח

(*)4 המשפחה לא מאכזבת וממתינה עד שהAIRPORT יגמר

הישות עברה את המסלול בצורה נכון.

#	time	name	key	value
1	8.95209	family0	type	3.000000
2	9.50220	family0	table	18.000000
3	202.20548	family0	Satiety_level	4.563472
4	205.30364	family0	disappointed	0.000000

נספח 3 – התפלגויות סימולציה, תבונות ומשאבים

נספח 3.1 – זמני הגעה

התפלגות	ישות
$interarrivals \sim Exp(\lambda)$	משפחה
3 נפשות – <input type="radio"/> 33%	
4 נפשות – <input type="radio"/> 40%	
5 נפשות – <input type="radio"/> 27%	
$interarrivals \sim Exp(\lambda)$	זוג
$bus\ time \sim U_c(60,75)$	קבוצת היחידים $Quantity \sim U_d(100, 150)$

נספח 3.2 – זמני שירות

התפלגות	שירות
$Norm(0.5, 0.2)$	קבלת מס' שולחן
$Norm(1.5, 0.7)$	הכנת מנה בעמדה
$Exp(2)$	קבלת משקה (בר חיצוני)
$Exp(2.4)$	קבלת משקה (בר פנימי)

נספח 3.3 – זמני פעולה

התפלגות	פעולה
$Triangle(3,5,4)$	מציאת חניה
$Exp(0.8)$	זמן אכילת מנה (קבלת פנים)
$Exp(2/7)$	זמן חיפוש מקום באולם
$Norm(12,3)$	זמן אכילת מנה בשירות
$Norm(7,4)$	זמן אכילת מנה צמחונית
$U_c(2.5,4)$	זמן אכילת קינוח
$Triangle(5,10,7)$	זמן ריקוד

נספח 3.4 – התפלגויות נוספות

התפלגות	מאפיין
$U_d(1,95)$	מס' שולחן
$U_c(140,160)$	תחילת החופה
$U_c(20,35)$	משך החופה

נספח 3.4 – טבלת ישוות

שם המשאב	קיבולת (Capacity)	אורך תור מקסימלי (queue_size)
מתחם החניה (parking)	Inf	Inf
עמדת קבלה 1 (reception1)	2	Inf
עמדת קבלה 2 (reception2)	2	Inf
עמדת פוקאצ'יות (focaccias)	3	Inf
עמדת באנים (buns)	3	Inf
עמדת טורטיות (tortillas)	4	Inf
עמדת הツמחונית (vegs)	4	Inf
עמדת סושי 1 (sushi1)	1	Inf
עמדת סושי 2 (sushi2)	2	Inf
בר חיצוני (externalBar)	3	Inf
חופה (huppa)	0	Inf
בר פנימי 1 (internalBar1)	5	Inf
בר פנימי 2 (internalBar2)	7	Inf
עמדת קינוחים (desserts)	8	7

הקוד לייצור הישויות

```
weddingSimulation<- simmer ("sim name")%<%
add_resource ("reception1", capacity=2, queue_size=Inf)%<%
add_resource ("reception2", capacity=2, queue_size=Inf)%<%
add_resource ("focaccias", capacity=3, queue_size=Inf)%<%
add_resource ("buns", capacity=3, queue_size=Inf)%<%
add_resource ("sushi1", capacity=1,queue_size=Inf)%<%
add_resource ("sushi2", capacity=2,queue_size=Inf)%<%
add_resource ("vegs", capacity=4, queue_size=Inf)%<%
add_resource ("tortillas", capacity=4, queue_size=Inf)%<%
add_resource ("externalBar", capacity=3, queue_size=Inf)%<%
add_resource ("internalBar1", capacity=5,queue_size=Inf)%<%
add_resource ("internalBar2", capacity=7,queue_size=Inf)%<%
add_resource ("desserts", capacity=8, queue_size=7, preemptive = FALSE)%<%
add_resource ("parking", capacity=Inf, queue_size =Inf)%<%
add_resource ("huppa", capacity=0, queue_size =Inf)
```

נספח 3.5 – טבלת תכונות

יעוד	זוג	משפחה 3 נפשות	משפחה 4 נפשות	משפחה 5 נפשות	תמונה/סוג יישות	
1	2	3	4	5	סוג לקוח (type)	
		Ud(1,95)				
			Inf	3	מספר שולחן מאובצב (disappointed) רק אורח שעבר בכל המסלולים "ICHSHUB CMOROZA"	
Rnd(1,2,3,0,0)	Rnd(1,2,3,4,5)	Rnd(1,2,3,4,5)	Rnd(1,2,3,4,5)	Rnd(1,2,3,4,5)	סדר הליכה לדוכנים (1- טורטיה, 2- צמחוני, 3- סושי, 4- באנים, 5- פוקאצ'יות)	
[1:6]	[1:6]	[1:6]	[1:6]	[1:6]	Counter – סופר את מספר דוכני האובל שהישות עבירה בו. (6 בغال התנא יציאה)	

נספח 3.6 – קביעת רמת השובע של קבוצה

הקוד שבירצנו לקביעת רמת השובע

```
satietyGenerator <- function() { # satiety function
  selector <- runif(1)
  u <- runif(1)

  if (selector < 15/30) #f1
    ans = (7*u+1)^(1/3)
  else if (selector > 15/30 & selector < 21/30) #f2
    ans = sqrt(3*u+1)
  else if (selector > 21/30 & selector < 23/30) #f3
    ans = u+1
  else if (selector > 23/30 & selector < 28/30) #g
    ans = 20/6 - sqrt(16/9-5/3*u)
  else
    ans = 5 - 2*sqrt(1-u)

  return (ans)
}

groupsatiety <- function (n = 1) {
  return (sum(replicate(n,satietyGenerator())))
}
```

נספח 3.7 – קביעת התפלגות הגעת הזוגות והמשפחות

על מנת לקבוע את התפלגות הגעת הזוגות והיחידים קראנו תחילה את נתונים המשפחות מהאקסל והמרנו את הנתונים משנהוות לדקות (הימרנו את הנתונים לפני כן באקסל מ-date למספר וייצגנו אותו בשניות).

```
### calculate and choose distribution for entire time of couples and families based on excel
#read data of family and couple
data <- read.csv(file.choose(), header = T)

diff_couples<-data$couples_diff/60 #converting to minutes from seconds
diff_families<-data$families_diff/60
```

לאחר מכן, בדקו ע"י פונקציית `fitdist` 3 התפלגויות, מעריכית, נורמלית ואחדית. לאחר מכן הוצאנו את מדדי AIC,BIC,LOGLIKLYWOOD summary על מנת לבדוק מה המודדים המינימליים או המקסימליים על פי הנדרש.

הקוד עבור הזוגות

```
> summary(couple_Fit_exp)
Fitting of the distribution ' exp ' by maximum likelihood
Parameters :
  estimate Std. Error
rate 0.5588291 0.07983248
Loglikelihood: -77.51367  AIC: 157.0273  BIC: 158.9192
> summary(couple_Fit_norm)
Fitting of the distribution ' norm ' by maximum likelihood
Parameters :
  estimate Std. Error
mean 1.789456 0.2618432
sd 1.832903 0.1851509
Loglikelihood: -99.21713  AIC: 202.4343  BIC: 206.2179
Correlation matrix:
  mean sd
mean 1 0
sd 0 1

> summary(couple_Fit_unif)
Fitting of the distribution ' unif ' by maximum likelihood
Parameters :
  estimate Std. Error
min 0 NA
max 8 NA
Loglikelihood: -101.8926  AIC: 207.7853  BIC: 211.5689
Correlation matrix:
[1] NA
```

הקוד עבור המשפחות

```
> summary(family_Fit_exp)
Fitting of the distribution ' exp ' by maximum likelihood
Parameters :
  estimate Std. Error
rate 0.8119304 0.1159899
Loglikelihood: -59.20869  AIC: 120.4174  BIC: 122.3092
> summary(family_Fit_norm)
Fitting of the distribution ' norm ' by maximum likelihood
Parameters :
  estimate Std. Error
mean 1.231633 0.1882064
sd 1.317445 0.1330817
Loglikelihood: -83.03699  AIC: 170.074  BIC: 173.8576
Correlation matrix:
  mean sd
mean 1 0
sd 0 1

> summary(family_Fit_unif)
Fitting of the distribution ' unif ' by maximum likelihood
Parameters :
  estimate Std. Error
min 0.000000 NA
max 6.183333 NA
Loglikelihood: -89.27102  AIC: 182.542  BIC: 186.3257
Correlation matrix:
[1] NA
```

דוגמא לעוד מבחן KS שביצעו על הזוגות

```
> # KS test
> ks.test(diff_couples, 'pexp')

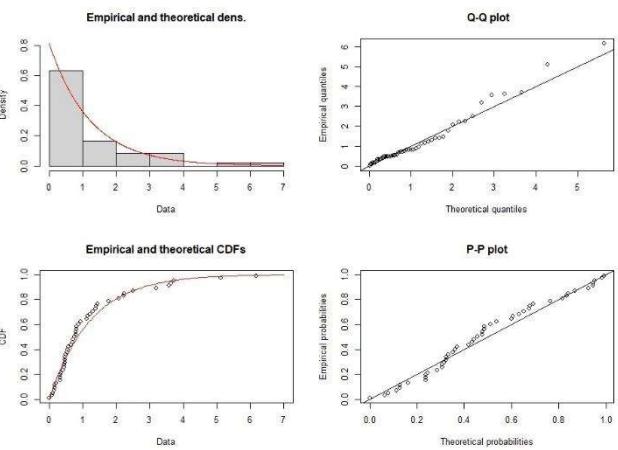
Asymptotic one-sample Kolmogorov-Smirnov test

data: diff_couples
D = 0.22328, p-value = 0.01511
alternative hypothesis: two-sided
```

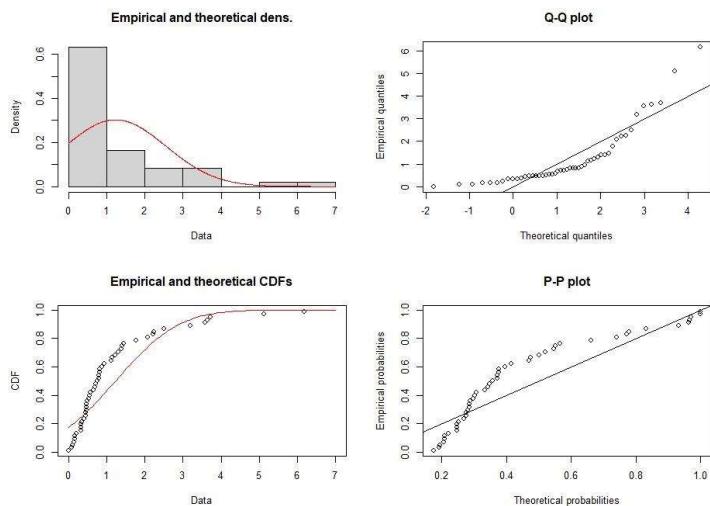
בנוסף, הוצאנו plot לכל התפלגות על מנת לקבל המבישה ויזואלית.

התפלגות המשפחות

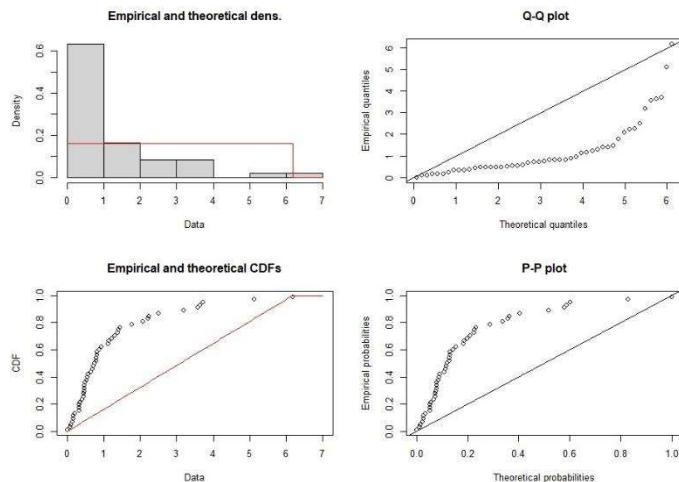
מעריצית



נורמלית

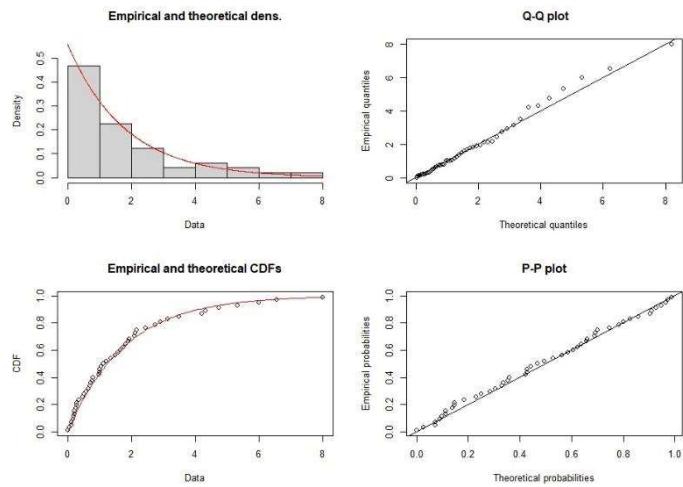


אחדית

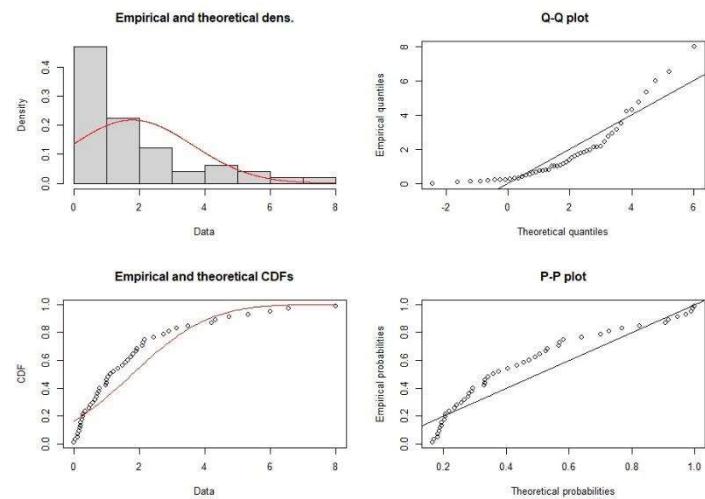


התפלגויות הזוגות

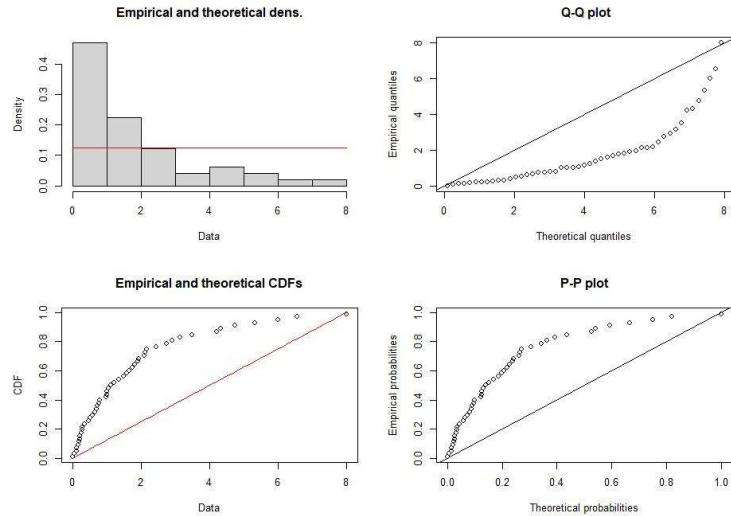
מעריצית



נורמלית



אחדת



ניתן לראות שההתפלגות המעריכית נמצאה עדיפה.

פלט לבחירת התפלגות עדיפה:

```
> print(paste("Best distribution for couples is EXP with lambda:",round(couple_fit_exp$estimate[1], 5)))
[1] "Best distribution for couples is EXP with lambda: 0.55883"
> lambda_couple <- as.numeric(couple_fit_exp$estimate[1])
> print(paste("Best distribution for families is EXP with lambda:",round(family_fit_exp$estimate[1], 5)))
[1] "Best distribution for families is EXP with lambda: 0.81193"
> lambda_family <- as.numeric(family_fit_exp$estimate[1])
```

נספח 4 – חישוב המדדים שבחרנו

4.1. חישוב זמן המתנה ממוצע בעמדות קבלת הפנים (עמדות האוכל בלבד) :

בעזרת פונקציות מתאימות ושימוש בשאליות חישבנו זמן המתנה ממוצע כולל עבור עמדות האוכל בלבד.

```
# Arrivals Table
fullData <- get_mon_arrivals(mm1envs,ongoing = F, per_resource = T)%>%
  mutate(waiting_time = end_time - start_time - activity_time)
#Resource Table
resourceData <- get_mon_resources(mm1envs)
# Attribute Table
attributeData <- get_mon_attributes(mm1envs)

# Measure 1: Average waiting time
Waiting_AVG <- sqldf("SELECT replication, AVG(waiting_time)
                       FROM fullData
                       WHERE resource NOT IN ('huppa', 'externalBar', 'internalBar1',
                                              'internalBar2', 'desserts', 'parking', 'reception1', 'reception2')
                       GROUP BY replication
                       ORDER BY 1 ASC")
Measure1 <- Waiting_AVG$`AVG(waiting_time)`
```

4.2. חישוב אחוז המאוכזבים באירוע :

כדי לחשב את אחוז המאוכזבים באירוע, תחילה חישבנו את מספר המאוכזבים באירוע (disaGuests). לאחר מכן, חילקו במספר הכלול של אורחים הגינו לחתונה (finishGuests). האחוז המתקבל הינו (Mesure2).

```
# Measure 2: % of disappointed guests
finishGuests <- sqldf("SELECT replication, COUNT(*) as finished
                        FROM attributeData
                        WHERE key = 'disappointed'
                        GROUP BY replication
                        ORDER BY 1 ASC")
disaGuests <- sqldf("SELECT replication, COUNT(*) as disa
                        FROM attributeData
                        WHERE key = 'disappointed' and value = 1
                        GROUP BY replication
                        ORDER BY 1 ASC")
Measure2 <- disaGuests$disa/finishGuests$finished
```

4.3. חישוב אחוז הנוכחים בחופה:

תחליה חישבנו את מספר האורחים שנכנסו בחופה (arrivedHuppaOnTime) . לאחר מכן, חישבנו את מספר המוזמנים הכלל שהגיעו לארוע (מספר היעילות שנוצרו) (inviteGuests).

```
# Measure 3: % of guests arrived to huppa in time

invitedGuests <- sqldf("SELECT replication, COUNT(*) as invited
                        FROM attributeData
                        WHERE key = 'table'
                        GROUP BY replication
                        ORDER BY 1 ASC")
arrivedHuppaOnTime <- sqldf("SELECT replication, MAX(queue) as arrived
                                FROM resourceData
                                WHERE resource = 'huppa'
                                ORDER BY 1 ASC")

Measure3 <- arrivedHuppaOnTime$arrived/invitedGuests$invited
```

נספח 5 – חישוב מספר החזרות הנדרש להערכת הסימולציה

5.1. חישוב הממוצעים וסטיות התקן

עבור המצב הנוכחי:

```
# Mean & SD for all the Measures

mean_Measure1 <- mean(Measure1)
sd_Measure1 <- sd(Measure1)

mean_Measure2 <- mean(Measure2)
sd_Measure2 <- sd(Measure2)

mean_Measure3 <- mean(Measure3)
sd_Measure3 <- sd(Measure3)
```

mean_Measure1	17.904705654472
mean_Measure2	0.759424673121728
mean_Measure3	0.327648124892951
sd_Measure1	2.57233434824055
sd_Measure2	0.0104713230994785
sd_Measure3	0.0194201196793479

עבור חלופה 1 – 2 ממצאים + עמדת פירות

```
# Mean & SD for all the Measures|
```

```
mean_Measure1_halufa1 <- mean(Measure1)
sd_Measure1_halufa1 <- sd(Measure1)

mean_Measure2_halufa1 <- mean(Measure2)
sd_Measure2_halufa1 <- sd(Measure2)

mean_Measure3_halufa1 <- mean(Measure3)
sd_Measure3_halufa1 <- sd(Measure3)
```

mean_Measure1_halufa1	12.7448886127162
mean_Measure2_halufa1	0.233914732964883
mean_Measure3_halufa1	0.387684580694828
sd_Measure1_halufa1	2.09986691561952
sd_Measure2_halufa1	0.0145951377258496
sd_Measure3_halufa1	0.018495654355335

עבור חלופה 2 – 2 ממצאים + הוספת עמדת טורטיות

```
# Mean & SD for all the Measures|
```

```
mean_Measure1_halufa2 <- mean(Measure1)
sd_Measure1_halufa2 <- sd(Measure1)

mean_Measure2_halufa2 <- mean(Measure2)
sd_Measure2_halufa2 <- sd(Measure2)

mean_Measure3_halufa2 <- mean(Measure3)
sd_Measure3_halufa2 <- sd(Measure3)
```

mean_Measure1_halufa2	15.5462457436415
mean_Measure2_halufa2	0.247449046529787
mean_Measure3_halufa2	0.428531172164386
sd_Measure1_halufa2	3.0313910494402
sd_Measure2_halufa2	0.017719509503177
sd_Measure3_halufa2	0.0239517193697673

5.2. חישוב הדיק היחסי ומספר החזרות הנדרש

הקוד בו השתמשנו לחישוב

```
n0<-20
gamma_ <- 0.1
alfa_total <- 0.09
alfa_i <- alfa_total/3
t <- qt(1 - (alfa_i)/2,n0 -1)
gamma_tag <- gamma_/(1 + gamma_)

# calculate the relative accuracy
calc_relative_accuracy <- function(mean,sd){
  (t*sd/sqrt(n0))/mean
}

# calculate the rough number of needed iterations
number_of_replications <- function(relative_acuuracy){
  n0*(relative_acuuracy/gamma_tag)^2
}

# relative accuracy for each measure
relative_accuracy_measure1 <- calc_relative_accuracy(mean_Measure1, sd_Measure1)
relative_accuracy_measure2 <- calc_relative_accuracy(mean_Measure2, sd_Measure2)
relative_accuracy_measure3 <- calc_relative_accuracy(mean_Measure3, sd_Measure3)

# rough number of needed iterations for each measure
n_needed_measure1 <- number_of_replications(relative_accuracy_measure1)
n_needed_measure2 <- number_of_replications(relative_accuracy_measure2)
n_needed_measure3 <- number_of_replications(relative_accuracy_measure3)
```

alfa_i	0.03
alfa_total	0.09
gamma_	0.1
gamma_tag	0.0909090909090909
n0	20
t	2.34564753356275

פלט

עבור המצב הקיים

relative_accuracy_measure1	0.0753542908495564	n_needed_measure1	13.7414113416437
relative_accuracy_measure2	0.00723210317597905	n_needed_measure2	0.126574025562175
relative_accuracy_measure3	0.0310879418662007	n_needed_measure3	2.33883351333259

עבור חלופה 1

relative_accuracy_measure1...	0.0864178551080431	n_needed_measure1_halufa1	18.0726705491688
relative_accuracy_measure2...	0.0327264107646662	n_needed_measure2_halufa1	2.59186346692114
relative_accuracy_measure3...	0.0250229742236951	n_needed_measure3_halufa1	1.5152811583793

עבור חלופה 2

relative_accuracy_measure1...	0.102273752670063	n_needed_measure1_halufa2	25.3130075742256
relative_accuracy_measure2...	0.0375589694255789	n_needed_measure2_halufa2	3.413836366034
relative_accuracy_measure3...	0.029315819012963	n_needed_measure3_halufa2	2.07978973144994

5.3. בדיקה שמספר החזרות הנדרש אכן עומד בבדיקה היחסית

ניתן לראות כי הבדיקה היחסית בכל המצבים קטן מ - 0.090909

עבור המצב הקיים

n0	30
relative_accuracy_measure1	0.0359519449264361
relative_accuracy_measure2	0.0218447920326684
relative_accuracy_measure3	0.0230543557029554

עבור חלופה 1

n0	30
relative_accuracy_measure1_halu...	0.044025479004291
relative_accuracy_measure2_halu...	0.0502257540894442
relative_accuracy_measure3_halu...	0.0189756076684734

עבור חלופה 2

n0	30
relative_accuracy_measure1_halu...	0.0778391999862596
relative_accuracy_measure2_halu...	0.0383026674503346
relative_accuracy_measure3_halu...	0.0216627079081773

5.4. תוצאות המדדים לאחר הרצת מספר החזרות הנדרש

עבור הממצב המקורי

n0	30
mean.Measure1	25.7380893488757
mean.Measure2	0.221311498180915
mean.Measure3	0.133228293080554
sd.Measure1	2.43515981013485
sd.Measure2	0.0161370213342675
sd.Measure3	0.00763803560337609

עבור חלופה 1

n0	30
mean.Measure1_halufa1	13.9672074185124
mean.Measure2_halufa1	0.236973934779084
mean.Measure3_halufa1	0.398021994020706
sd.Measure1_halufa1	3.21295552699908
sd.Measure2_halufa1	0.0175727997514529
sd.Measure3_halufa1	0.0228454835058406

עבור חלופה 2

n0	30
mean.Measure1_halufa2	14.3843263746633
mean.Measure2_halufa2	0.240035693749333
mean.Measure3_halufa2	0.450154032640443
sd.Measure1_halufa2	2.84708509035594
sd.Measure2_halufa2	0.0175401330961029
sd.Measure3_halufa2	0.0271421477667405

5.5. השוואת המודלים (מבחני t מדווגים)

הקוד בו השתמשנו

Measure 1 Comparison

```
measure1_original_halufa1 <- t.test(original$Measure1, halufa1$Measure1, alternative = "two.sided",
                                         paired = TRUE, var.equal = TRUE, conf.level = 0.97)
```

```

print(measure1_original_halufa1)

measure1_original_halufa2 <- t.test(original$Measure1, halufa2$Measure1, alternative =
"two.sided",
                                    paired = TRUE, var.equal = TRUE, conf.level = 0.97)

print(measure1_original_halufa2)

measure1_halufa1_halufa2 <- t.test(halufa1$Measure1, halufa2$Measure1, alternative =
"two.sided",
                                    paired = TRUE, var.equal = TRUE, conf.level = 0.97)

print(measure1_halufa1_halufa2)

# Measure 2 Comparison

measure2_original_halufa1 <- t.test(original$Measure2, halufa1$Measure2, alternative =
"two.sided",
                                    paired = TRUE, var.equal = TRUE, conf.level = 0.97)

print(measure2_original_halufa1)

measure2_original_halufa2 <- t.test(original$Measure2, halufa2$Measure2, alternative =
"two.sided",
                                    paired = TRUE, var.equal = TRUE, conf.level = 0.97)

print(measure2_original_halufa2)

measure2_halufa1_halufa2 <- t.test(halufa1$Measure2, halufa2$Measure2, alternative =
"two.sided",
                                    paired = TRUE, var.equal = TRUE, conf.level = 0.97)

print(measure2_halufa1_halufa2)

```

Measure 3 Comparison

```
measure3_original_halufa1 <- t.test(original$Measure3, halufa1$Measure3, alternative =  
"two.sided",  
                                     paired = TRUE, var.equal = TRUE, conf.level = 0.97)  
print(measure3_original_halufa1)  
  
measure3_original_halufa2 <- t.test(original$Measure3, halufa2$Measure3, alternative =  
"two.sided",  
                                     paired = TRUE, var.equal = TRUE, conf.level = 0.97)  
print(measure3_original_halufa2)  
  
measure3_halufa1_halufa2 <- t.test(halufa1$Measure3, halufa2$Measure3, alternative =  
"two.sided",  
                                     paired = TRUE, var.equal = TRUE, conf.level = 0.97)  
print(measure3_halufa1_halufa2)
```

אחד הפלטים

```
> # Measure 3 compression Halufa 1 and Halufa 2  
> print(measure3_halufa1_halufa2)  
  
Paired t-test  
  
data: halufa1$Measure3 and halufa2$Measure3  
t = -10.851, df = 29, p-value = 1.003e-11  
alternative hypothesis: true mean difference is not equal to 0  
97 percent confidence interval:  
-0.06309663 -0.04116745  
sample estimates:  
mean difference  
-0.05213204
```

```
##----- 1. all functions -----
```

```
addService<- function (path,sname,timeDist){ # catch a source
  updatedPath <- seize(path, sname)%>%
    timeout(timeDist)%>%
    release(sname)
  return(updatedPath)
}
```

```
trimmedNorm<-function (mu,sd){ # trimmed norm
  while(TRUE){
    sample<-rnorm(1,mu,sd)
    if (sample>0)
      return (sample)
  }
}
```

```
# choose favorite food for guest
favouriteFoods <- function() {
  y = 0
  if (get_attribute(weddingSimulation,"type") == 1){ #single will save a place for main dishes
    y <- sample(partial_foods)
  }
  else {y <- sample(foods)} #couple or family
  return (y)
}
```

```
# return type of guest by number of guests
get_type <- function(){
```

```

name<- get_name(weddingSimulation)

x <- 0

if (substr(name,0,1) == "f"){ # family
  x <- rdiscrete(1,c(0.33,0.4,0.27), c(3,4,5))

}

else if (substr(name,0,1) == "c") {x <- 2} #couple
else {x <- 1} #single

return (c(x))

}

chanceForBarByType <- function(){ #return the percentage for going to the bar

y = 0.0

if (get_attribute(weddingSimulation,"type") == 1) {y <- 0.8} #single
else if (get_attribute(weddingSimulation,"type") == 2) {y <- 0.6} #couple
else {y <- 0.4} #family

return (y)

}

getType <- function(){ # an auxiliary function that helps with routing when entering the hall

y = 0

if (get_attribute(weddingSimulation,"type") > 2) {y <- 0} #family
else {y <- 1} #single or couple

return (y)

}

get_food <- function(){ #return for each food attribute the number of road plan to go

counter <- get_attribute(weddingSimulation,"counter")

food <- as.integer(get_attribute(weddingSimulation,paste("food",counter,sep="")))

return (food)

}

```

```

groupEatingTime<-function(){ #The function samples the eating times of all
#members of the family/couple/individual and returns
#the highest eating time among all members of the group
familyMembers<- get_attribute(weddingSimulation,"type") # number of guest in the group
t<- rdiscrete(familyMembers,c(0.7,0.3),c(1,2)) # with a probability of 70 percent -
#a meat dish and with a probability of
#30 percent - a vegetarian dish
x<-c(0,0,0,0,0)
n=1
for(i in t){
  if(i==1) {x[n]<-trimmedNorm(12,3)} #meat
  else {x[n]<-trimmedNorm(7,4)} #vegetarian
  n=n+1
}
maxTime <-max(x) #the highest from group's member
return(maxTime)
}

satietyGenerator <- function() { # satiety function
  selector <- runif(1)
  u <- runif(1)

  if (selector < 15/30) #f1
    ans = (7*u+1)^(1/3)
  else if (selector > 15/30 & selector < 21/30) #f2
    ans = sqrt(3*u+1)
  else if (selector > 21/30 & selector < 23/30) #f3
    ans = u+1
  else if (selector > 23/30 & selector < 28/30) #g
    ans = 20/6 - sqrt(16/9-5/3*u)
  else
}

```

```

ans = 5 - 2*sqrt(1-u)

return (ans)
}

groupSatiety <- function (n = 1) {
  return (sum(replicate(n,satietyGenerator())))
}

returnSatiety<- function(){
  x<-get_attribute(weddingSimulation,"Satiety_level")
  y <- 0
  if (x < 4.5) {y <- 1}
  return (c(y,y,TRUE))
}

##----- 2. all simulation parameters -----

simulationTime <- 6*60
foods <- c(1,2,3,4,5) # c("tortilla","veg","sushi","bun","focaccia")
partial_foods <- c(1,2,3,0,0)

##----- 3. Init Simulation and add all resources -----
-----


weddingSimulation<- simmer("sim name")%>%
  add_resource("reception1",capacity=2,queue_size=Inf)%>%
  add_resource("reception2",capacity=2,queue_size=Inf)%>%
  add_resource("focaccias",capacity=3,queue_size=Inf)%>%
  add_resource("buns",capacity=3,queue_size=Inf)%>%

```

```

add_resource("sushi1",capacity=1,queue_size=Inf)%>%
add_resource("sushi2",capacity=2,queue_size=Inf)%>%
add_resource("vegs",capacity=4,queue_size=Inf)%>%
add_resource("tortillas",capacity=4,queue_size=Inf)%>%
add_resource("externalBar",capacity=3,queue_size=Inf)%>%
add_resource("internalBar1",capacity=5,queue_size=Inf)%>%
add_resource("internalBar2",capacity=7,queue_size=Inf)%>%
add_resource("desserts",capacity=8,queue_size=7, preemptive = FALSE)%>%
add_resource("parking",capacity=Inf,queue_size =Inf)%>%
add_resource("huppa",capacity=0,queue_size =Inf)

##----- 4. All trajectories, start from main trajectory and add sub-trajectories ABOVE IT . -----
dessertPath<- trajectory("dessertPath")%>%
# log_("I catch the dessert")%>%
timeout(function() runif(1,2.5,4))%>% #people eat in dessert area
release("desserts",1)%>%
set_attribute("disappointed", 0) # not disappointed

dancePath <- trajectory("dancePath")%>%
# log_("im dancing")%>%
timeout(function() rtriangle(1,5, 10, 7)) # dancing

internalBarPath<-trajectory("internalBarPath")%>%
simmer::select(resources=c("internalBar1","internalBar2"), policy=c("shortest-queue"))%>%
#log_("im in internalBar")%>%
seize_selected(amount = 1) %>%
timeout(rexp(1,2.4)) %>%
release_selected(amount = 1)

externalBar <- trajectory("externalBar")%>%

```

```

#log_("Im drinking at external bar")%>%
addService("externalBar", function () rexp(1,2))

bunsPath<-trajectory("bunsPath")%>%
#log_("im in bunPath")%>%
addService("buns", function() trimmedNorm(1.5, 0.7))%>%
timeout(rexp(1,0.8))%>% #time for eating the food
branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()),c(0,1)),
continue = c(TRUE) ,externalBar)%>%
timeout(0)

focacciaPath<-trajectory("focacciaPath")%>%
#log_("im in focacciaPath")%>%
addService("focaccias", function() trimmedNorm(1.5, 0.7))%>%
timeout(rexp(1,0.8))%>%
branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()),c(0,1)),
continue = c(TRUE) ,externalBar)%>%
timeout(0)

vegsPath<-trajectory("vegsPath")%>%
#log_("im in vegPath")%>%
addService("vegs", function() trimmedNorm(1.5, 0.7))%>%
timeout(rexp(1,0.8))%>%
branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()),c(0,1)),
continue = c(TRUE) ,externalBar)%>%
timeout(0)

tortillasPath<-trajectory("tortillasPath")%>%
#log_("im in tortillasPath")%>%
addService("tortillas", function() trimmedNorm(1.5, 0.7))%>%
timeout(rexp(1,0.8))%>%

```

```

branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()),c(0,1)),
continue = c(TRUE) ,externalBar)%>%
timeout(0)

sushiPath<-trajectory("sushiPath")%>%
simmer::select(resources=c("sushi1","sushi2"), policy=c("shortest-queue-available") )%>%
#log_("im in sushiPath")%>%
seize_selected(amount = 1) %>%
timeout(trimmedNorm(1.5, 0.7)) %>%
release_selected(amount = 1)%>%
timeout(rexp(1,0.8))%>%
branch (option = function() rdiscrete(1,c(1 - chanceForBarByType(),chanceForBarByType()))
,c(0,1)), continue = c(TRUE) ,externalBar)%>%
timeout(0)

reception<- trajectory("reception")%>%
simmer::select(resources=c("reception1","reception2"), policy=c("shortest-queue") )%>%
#log_("im in reception")%>%
seize_selected(amount = 1) %>%
timeout(trimmedNorm(0.5, 0.2)) %>%
release_selected(amount = 1)%>%
leave(0.07)%>%
set_attribute("table", function () floor(runif(1,1,96)))

mainPath<- trajectory("mainPath")%>%
set_attribute("type" , function () get_type())%>%
join(reception)%>%
#log_("Im finish the reception")%>%
set_attribute("disappointed", 1)%>% # disappointed
clone(function() get_attribute(weddingSimulation, "type"))%>%
set_attribute(keys=c("food1","food2","food3","food4","food5"), values = function()
favouriteFoods())%>%

```

```

set_attribute("counter", 1)%>%
branch(option = function() get_food(), continue = c(TRUE,TRUE,TRUE,TRUE,TRUE),
       tortillasPath,vegsPath,sushiPath,bunsPath,focacciaPath)%>% #split to food stands
set_attribute("counter", 1, mod = "+")%>%
rollback(times = 4, amount = 2)%>%
# log_("finish dishes")%>%
synchronize(wait = TRUE, mon_all = TRUE)%>%
#log_("finish all dishes")%>%
seize("huppa",1)%>% # everyone waiting until the end of the huppa
release("huppa",1)%>% # huppa end
# log_("huppa is over")%>%
branch (option = function() getType(), continue = c(TRUE),
         internalBarPath)%>% # sending couples and singles to the internal bar
timeout(rexp(1,1/3.5))%>% # time to search for table
# log_("arrived to table")%>%
timeout(function() groupEatingTime())%>%
# log_("finish to eat")%>%
set_attribute("Satiety_level", function()
groupSatiety(get_attribute(weddingSimulation,"type")))%>%
set_prioritization(values = function() returnSatiety())%>%
# log_("arrived to dessert queue")%>%
seize("desserts",1, continue = c(FALSE, TRUE), post.seize = dessertPath,
      reject = dancePath)%>% #if catch dessert go to eat
rollback(times = 2, amount = 1)%>%
leave(1)

parkingPath <- trajectory("parkingPath") %>%
#log_("im in parking")%>%
addService("parking", function() rtriangle(1,3, 5, 4))%>%
join(mainPath)

```

```

trigger<- trajectory() %>%
  timeout_from_global("busTime")%>%
  activate("single")

resetPath <- trajectory("resetPath")%>%
  set_global(keys = "busTime", value = function() runif(1,60,75))%>%
  set_global(keys = "singlesQunty", value = function() floor(runif(1,100,151)))%>%
  set_global(keys = "weddingTime", value = function() runif(1,140,160))%>%
  set_global(keys = "weddingLength", value = function() runif(1,20,35))%>%
  set_global(keys = "huppaEnd", value = function()
    get_global(weddingSimulation,key="weddingTime") + get_global(weddingSimulation,
    key="weddingLength"))%>%
  set_global(keys = "barClose", value = function() get_global(weddingSimulation,key="weddingTime")
  - 10)%>%
  set_queue_size("reception1",Inf)%>%set_capacity("reception1", 2)%>%
  set_queue_size("reception2",Inf)%>%set_capacity("reception2", 2)%>%
  set_queue_size("focaccias",Inf)%>%set_capacity("focaccias", 3)%>%
  set_queue_size("buns",Inf)%>%set_capacity("buns", 3)%>%
  set_queue_size("sushi1",Inf)%>%set_capacity("sushi1", 1)%>%
  set_queue_size("sushi2",Inf)%>%set_capacity("sushi2", 2)%>%
  set_queue_size("vegs",Inf)%>%set_capacity("vegs", 4)%>%
  set_queue_size("tortillas",Inf)%>%set_capacity("tortillas", 4)%>%
  set_queue_size("externalBar",Inf)%>%set_capacity("externalBar", 3)%>%
  set_queue_size("internalBar1",Inf)%>%set_capacity("internalBar1", 5)%>%
  set_queue_size("internalBar2",Inf)%>%set_capacity("internalBar2", 7)%>%
  set_queue_size("desserts",7)%>%set_capacity("desserts", 8)%>%
  set_queue_size("parking",Inf)%>%set_capacity("parking", Inf)%>%
  set_queue_size("huppa",Inf)%>%set_capacity("huppa", 0)%>%
  timeout_from_global("barClose")%>%
  set_queue_size("externalBar",0)%>%
  timeout(10)%>%

```

```

set_queue_size("focaccias",0)%>%
set_queue_size("buns",0)%>%
set_queue_size("sushi1",0)%>%
set_queue_size("sushi2",0)%>%
set_queue_size("vegs",0)%>%
set_queue_size("tortillas",0)%>%
timeout_from_global("weddingLength")%>%
set_capacity("huppa", Inf)

```

##----- 5. All Generators, ALWAYS LAST. -----

--

```

weddingSimulation%>%
add_generator("reset", resetPath, distribution = at(0))%>%
add_generator("couple", parkingPath, distribution = to(4*60, function () rexp(1,lambda_couple)),
mon=2)%>%
add_generator("family", parkingPath, distribution = to(4*60, function () rexp(1,lambda_family)),
mon=2)%>%
add_generator("single", mainPath, when_activated(function() get_global(weddingSimulation,
"singlesQunty")), mon=2) %>%
add_generator("trigger", trigger, at(0))

```

##----- 6. reset, run, plots, outputs -----

```

mm1envs <- mclapply(1:30, function(i){
  set.seed(455+i)
  reset(weddingSimulation)%>%run(until=simulationTime)%>%
  wrap()
})

```