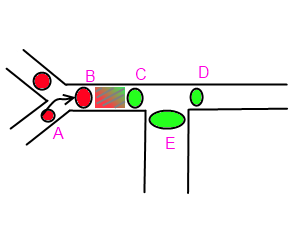
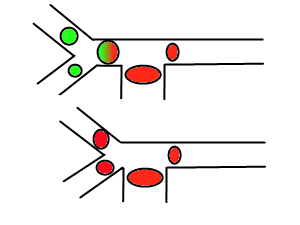
**2)**

ما بمنظور پیاده سازی ایده خود زمین و راه‌های آن را که معرّف نقشه بازی است، به یک گراف مدل کردیم. در این مدل راس‌ها به محل‌های ورود و خروج اطلاق میشود. به عنوان مثال در **شکل روبرو(**شکل 1**)** نقاط پر رنگ نشان دهنده یک راس در گراف هستند ( چراکه محل ورود یا خروج یک مسیر جدید می‌باشند ) و خطوط درون شکل نشان دهنده یالها میباشند. در این گراف، یال AB نشان میدهد که یک مسیر از رأس A به B و بالعکس وجود دارد که نیاز به عبور از هیچ رأس دیگری ندارد. استفاده از این شیوه مدل سازی به ما کمک می‌کند که مستقل از مختصات جغرافیایی فرد در حال تعقیب ( بدون توجه به x و y مکانی ) و تنها به کمک آگاهی‌ از چند رأس آخری که از آن‌ها عبور کرده ، محل نسبی‌ فرد را تشخیص دهیم. همچنین به کمک روشی که در زیر معرفی‌ میشود ، ما قادر به تشخیص تغییر مسیرها خواهیم بود. بدین منظور ما گراف را به کمک ماتریس مجاورت (Adjacent Matrix) پیاده سازی کردیم. در نتیجه، اگر فرض کنیم  که می‌دانیم فرد در حال تعقیب به ترتیب از راس های A و B عبور کرده باشد، مشخص میشود که این فرد در حال حرکت بسمت راس C می‌باشد و در ناحیه‌‌ هاشور خورده مشخص شده در شکل میباشد. اینکار ( تشخیص مکان بعدی و در نتیجه آن، محل نسبی فرد) بکمک تفریق مجموعه رئوس مجاور B از مجموعه رئوس مجاور راس A بدست می آید، که مشخصا پردازش بسیاری کمی‌ را نیاز دارد). از آنجا که در مدل پیشنهادی ما، رئوس بعبارتی یک گذرگاه (trigger) می باشند، ما از نگرش اتفاق محور (Event Base) در انجام محاسبات فوق استفاده میکنیم. به این معنا که در هر بار عبور کردن فرد در حال تعقیب از یک گذرگاه، محاسبات فوق یکبار انجام میشود.   
از آنجا که ما مسیر پیشروی فرد در دست تعقیب را بکمک ۲ راس آخری که وی از آن ها عبور کرده است ، محاسبه ‌کنیم، این نکته مهم است که این ۲ راس ( یا گذرگاه) باید متوالی باشند. از طرف دیگر از آنجا که یکی‌ از جنبه‌هایی که در این طرح پیشنهادی مبنا بوده است Real بودن آن است، ما تنها در صورتی‌ event عبور از یک راس را پردازش می‌کنیم که یکی‌ از افراد در حال تعقیب، شخص در حال فرار را در آن لحظه ببیند.  
به عنوان مثال اگر فرد در دست تعقیب، بعد از عبور از گذرگاه‌های A، B و C ، از گذرگاه E عبور کند ولی‌ نیرویی به او نرسد و مدتی‌ بعد در حال عبور از گذرگاه D دیده شود، نشان میدهد که او مدتی‌ توسط شخصی‌ در حال تعقیب نبود و ما اطلاعاتی‌ از وی نداشتیم. پس نمیتوانیم در تنها بر اساس راس D نظر بدهیم. در اینجا ما بر اساس نقشه واقعی بازی ( و نه مدل گراف آن) مسیر بعدی را بدست می آوریم. اینکار میتواند بشکل های مختلف انجام گیرد که ما از روش شبیه سازی ادامه حرکت با سرعت بسیار بالا استفاده کردیم .  
همچنین ما در ادامه سعی‌ کردیم برای بهینه سازی گراف ، تغیراتی‌ در انتخاب رئوس داشته باشیم. بدین منظور ما سعی کردیم بر اساس فاصله دو راس متوالی و اهمیت ناحیه‌‌ بین  ۲ راس ( که نماینده یک یال در گراف، و یک مسیر در نقشه میباشد) رئوس را با یکدیگر ادغام کنیم. در **شکل زیر** (شکل2) دو نمونه از نحوه بهینه انتخاب رئوس **شکل قبل(1)** آورده شده است.



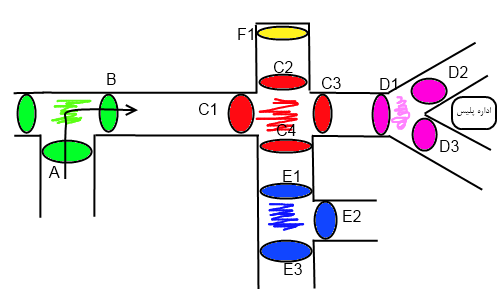
شکل 1



شکل2

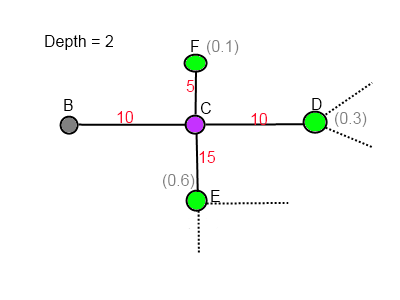
**درخت پیشبینی مسیر :**

همانطور که در بالا به آن اشاره شد، در هر بار عبور از رئوس ، ما یک پردازش برای برسی‌ محل بعدی فرد در حال گریز انجام میدهیم. ما همچنین همزمان با بروز رسانی اطلاعات فوق، یک درخت پیشبینی‌ مسیر نیز تشکیل میدهیم. علت همزمانی این دو عملیات، عدم امکان تغییر مسیر در بین دو راس است. یعنی‌ پس از اینکه فرد در دست تعقیب از یک گذرگاه عبور کرد، تا رسیدن فرد به رأس بعدی ، تغییر مسیری نمیتواند داشته باشد که این فرصت مناسبی برای تشکیل درخت ما مهیا می‌کند. (لازم به ذکر است که گذر از رئوس که منجر به ورود به یک ناحیه‌‌ چند راهی‌ میشود بشکل دیگری بررسی‌ میشود که از بیان این حالت خاص صرفنظر میکنیم). هدف از تشکیل درخت فوق که بنوعی از گراف اصلی ما ولی‌ بشکل خاص استفاده می‌کند ، به دست آوردن احتمال حرکت فرد در حال گریز به محل‌های در پیش روی اوست. نتیجه ای که از این درخت برای ما اهمیت دارد، بدست آوردن مکان‌های مناسب برای قرار دادن نیروها و بستن راه‌ها است.

برای نمونه و بر اساس **شکل روبرو**(3) ، رئوس C1 تا C4 از نظر مفهومی‌ نمایانگر یک منطق قابل پوشش توسط نیروهای تعقیب کنند می‌باشد، پس ما در درخت خود، برای چنین رئوسی، یک راس را به عنوان نماینده برای قرار گیریی نیرو استفاده میکنیم.   


شکل 3

نحوه تشکیل این درخت را با یک مثال نشان میدهیم.  
اگر فرض کنیم فرد در دست تعقیب بترتیب از گذرگاه‌های A و B گذشته باشد ، برای عمق ۲، درختی بشکل روبرو خواهیم داشت :



شکل 4

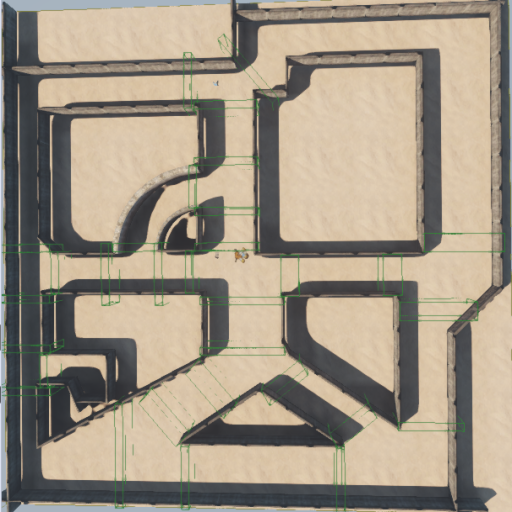
در این درخت وزن یالها نمایانگر طول مسیر و وزن برگ‌ها نشان دهنده احتمال آن مکان است. ( احتمال آنکه فرد درحال فرار به آنجا بگریزد )  
در محاسبه این احتمالات ما ۲ معیار اصلی‌ را مدنظر قرار دادیم :

1. میزان خطر گذرگاه ها :

این معیار توسط طراح بازی ارزش دهی‌ میشود و به شرایط فیزیکی یک راس و عامل‌های از این دست بستگی دارد. به عنوان مثال، یک مکان که دارای آیتم‌هایی برای پنهان شدن بازیکن باشد، نسبت به یک مسیر که به بن‌بست ختم میشود، مناسب تر در نظر گرفته میشود و از ارزش بالاتری برخوردار خواهد بود.

2. تاریخچه مسیر‌های انتخابی:  
در این قسمت، یک سیستم ارزش دهی جداگانه برای گذرگاه ها در نظر گرفته میشود که بر اساس مشاهدات نیروهای تعقیب کننده در حین تعقیب و بر اساس انتخاب های فرد در دست تعقیب، ارزش دهی میشود. به عنوان مثال، اگر بازیکن چندبار در یک دوراهی از مسیر سمت راست برود و اینکار توسط نیروها دیده شود، باعث میشود راس انتهای مسیر سمت راست از ارزش بیشتری نسبت به راس موجود در مسیر سمت چپ، برخوردار شود.

در زیر نمای نقشه ای که تیم ما آزمایشات خود را در آن انجام داده، در کنار گذرگاه های آن نمایش داده شده است :



شکل 5

**سیستم مرکزی تصمیم گیری :**