

آشنایی با شبکه های حسگر بی سیم

تهیه و تنظیم توسط سایت پارس بوک

www.ParsBook.org

شبکه حسگر چیست؟

شبکه حسگر/کارانداز () /کار¹ شبکه ای است متشکل از تعداد زیادی گره کوچک. /یا کارانداز وجود دارد. شبکه /کار بشدت با محیط فیزیکی تعامل دارد. از طریق حسگرها اطلاعات محیط را گرفته و از طریق کار انداز ها واکنش نشان می دهد. ارتباط بین گره ها بصورت بی سیم است. انسان کار میکند و نوعاً از لحاظ فیزیکی بسیار کوچک است و دارای محدودیت هایی در قدرت پردازش، ظرفیت حافظه، منبع تغذیه، ... می باشد. این محدودیت ها مشکلاتی را بوجود می آورد که منشأ بسیاری از مباحث پژوهشی مطرح در این زمینه است. این شبکه از پشته پروتکلی شبکه های سنتی پیروی می کند ولی بخاطر محدودیت ها و تفاوت های وابسته به کاربرد، و تکل ها باید باز نویسی شوند.

چرا شبکه های حسگر؟

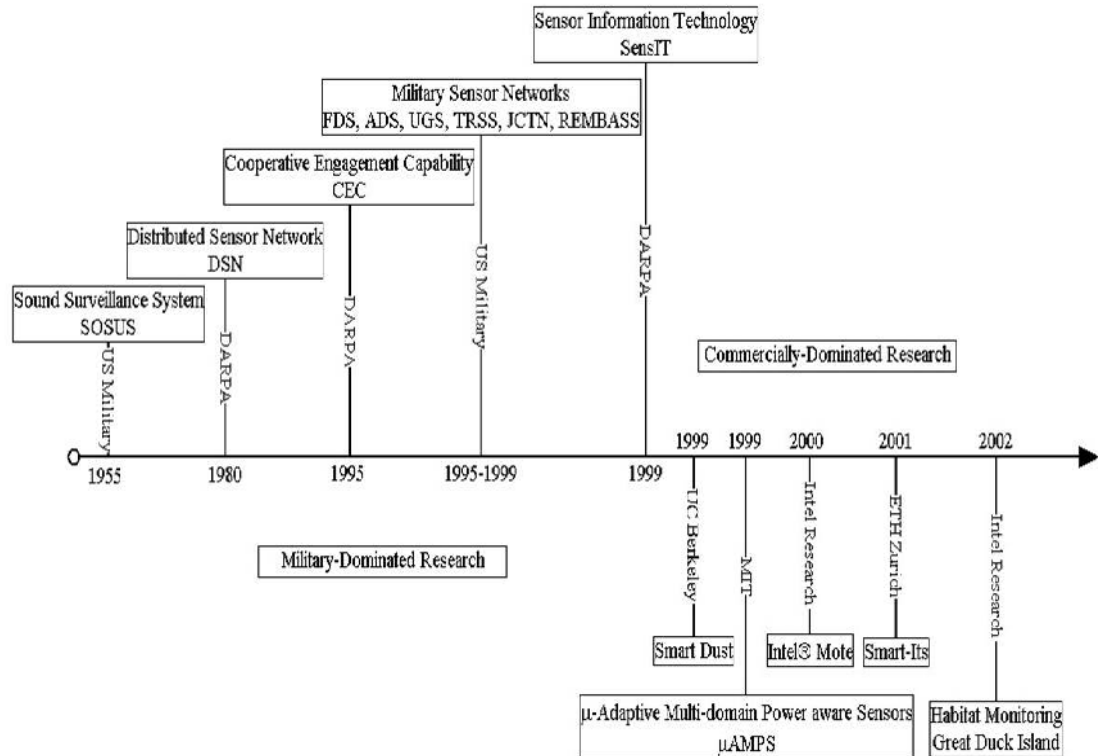
امروزه زندگی بدون ارتباطات بی سیم قابل تصور نیست. پیشرفت تکنولوژی CMOS و ایجاد مدارات کوچک و کوچکتر باعث شده است تا استفاده از مدارات بی سیم در اغلب وسایل الکترونیکی امروز ممکن شود. این پیشرفت همچنین باعث ریز حسگرها شده است. این ریز حسگرها توانایی انجام حس های بی شمار در کارهایی مانند شناسایی صدا برای حس کردن زلزله را دارا می باشند همچنین جمع آوری اطلاعات در مناطق دور افتاده و مکان هایی که برای اکتشافات انسانی مناسب نیستند را فراهم کرده است. اتومبیل ها می توانند از ریز حسگرهای بی سیم برای کنترل وضعیت موتور، فشار تایرها، ... استفاده کنند. خطوط مونتاژ می توانند از این سنسورها برای کنترل فرایند مراحل طول تولید استفاده کنند. در موقعیت های استراتژیک ریز حسگرها می توانند توسط هواپیما بر روی خطوط دشمن ریز برای رد گیری هدف (مانند ماشین یا انسان) . تفاوت اساسی این شبکه ها ارتباط آن با محیط و پدیده های فیزیکی است شبکه های سنتی ارتباط بین انسانها و پایگاه های اطلاعاتی را فراهم می کند در حالی که شبکه ی /کار مستقیماً با جهان فیزیکی در ارتباط است با استفاده از حسگرها محیط فیزیکی را مشاهده کرده، مشاهدات خود تصمیم گیری نموده و عملیات مناسب را انجام می دهند. نام شبکه حس /کار بی سیم یک نام عمومی است برای انواع مختلف که به منظورهای خاص طراحی می شود. برخلاف شبکه های سنتی که همه منظوره اند شبکه های /کار نوعاً تک منظوره هستند. در هر صورت شبکه های حسگر در نقاط مختلفی کاربرد دارند برخی از این کاربردها به :

- نظامی (برای مثال ردگیری اشیاء)
- (برای مثال کنترل علائم حیاتی)
- محیط (برای مثال آنالیز زیستگاه های طبیعی)
- صنعتی (برای مثال عیب یابی خط تولید)
- سرگرمی ()
- زندگی دیجیتال (برای مثال ردگیری مکان پارک ماشین)

¹ Wireless Sensor Actor Network (WSAN)

تاریخچه شبکه های حسگر

در شکل (1) طرح ها و ایده های اولیه شبکه های حسگر نشان داده شده است.



شکل (1) رخدادهای نگاری شبکه حسگر

اگرچه تاریخچه شبکه های حس/کار را به دوران جنگ سرد و ایده اولیه آن را به طراحان نظامی صنایع دفاع آمریکا نسبت می دهند ولی این ایده می توانسته در ذهن طراحان ربات های متحرک مستقل یا حتی طراحان شبکه های بی سیم موبایل نیز شکل گرفته باشد.

ساختار کلی شبکه حس/کار بی سیم

قبل از ارائه ساختار کلی ابتدا تعدادی از تعاریف کلیدی را ذکر می کنیم.

: وسیله ای که وجود شیئی رخدادهای یک وضعیت یا مقدار یک کمیت فیزیکی را تشخیص داده و به سیگنال الکتریکی تبدیل می کند.
 مغناطیس سنج و...
کارانداز: با تحریک الکتریکی یک عمل خاصی مانند باز و بسته کردن یک شیر یا قطع و وصل یک کلید را انجام می

: به گره ای گفته می شود که فقط شامل یک یا چند حسگر باشد.

گره کارانداز: به گره ای گفته می شود که فقط شامل یک یا چند کارانداز باشد.

کارانداز: به گره ای گفته می شود که مجهز به حسگر و کارانداز باشد.

شبکه حسگر: شبکه ای که فقط شامل گره های حسگر باشد. این شبکه نوع خاصی از شبکه حس/کار است.

کاربردهایی که هدف جمع آوری اطلاعات و تحقیق در مورد یک پدیده می باشد کاربرد دارد.

میدان حسگر/کارانداز: ناحیه کاری که گره های شبکه حس/کار در آن توزیع میشوند.

چاهک²: گرهی که جمع آوری داده ها را به عهده دارد. و ارتباط بین گره های حس/کار و گره مدیر وظیفه³ را برقرار می کند.

گره مدیر وظیفه: گرهی که یک شخصی بعنوان کاربر یا مدیر شبکه از طریق آن با شبکه ارتباط برقرار میکند. فرامین

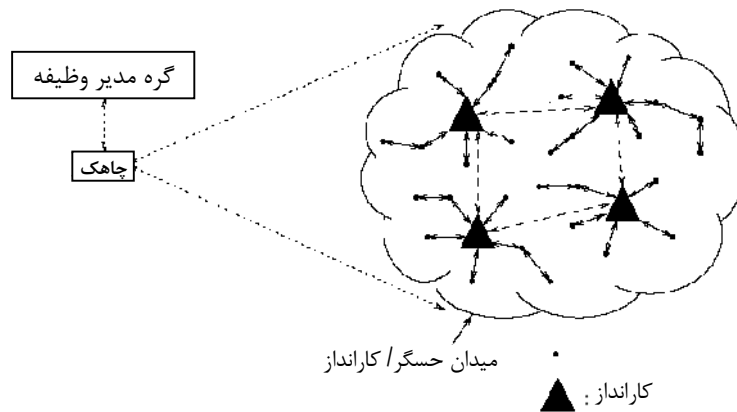
کنترلی و پرس و جو ها از این گره به شبکه ارسال شده و داده های جمع آوری شده به آن بر میگردد

شبکه حس/کار: شبکه ای متشکل از گره های حسگر و کارانداز یا حسگر/کارانداز است که حالت کلی شبکه های مورد بحث می باشد. به عبارت دیگر شبکه حس/کار شبکه ای است با تعداد زیادی گره که هر گره می تواند در حالت کلی دارای

تعدادی حسگر و تعدادی کارانداز باشد. در حالت خاص یک گره ممکن است فقط حسگر یا فقط کارانداز باشد.

ناحیه ای که میدان حس/کار نامیده می شود با چگالی زیاد پراکنده می شوند. یک چاهک پایش⁴ کل شبکه را بر عهده اطلاعات بوسیله چاهک جمع آوری می شود و فرامین از طریق چاهک منتشر می شود. شکل (2) را ببینید. مدیریت

وظایف میتواند متمرکز یا توزیع شده باشد. بسته به اینکه تصمیم گیری برای انجام واکنش در چه سطحی انجام شود دو ودکار و نیمه خودکار وجود دارد. که ترکیب آن نیز قابل استفاده است.



شکل (2) ساختار کلی شبکه حس/کار

ساختار خودکار: حسگرهایی که یک رخداد یا پدیده را تشخیص می دهند داده های دریافتی را به گره های کارانداز

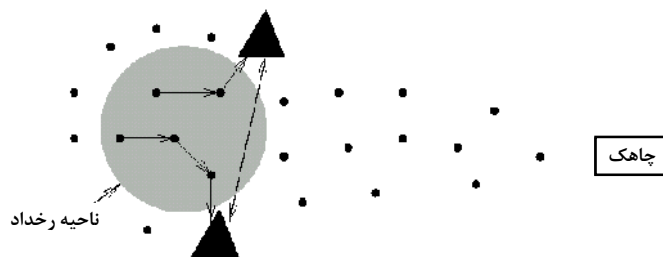
جهت پردازش و انجام واکنش مناسب ارسال می کنند. گره های کارانداز مجاور با هماهنگی با یکدیگر تصمیم گیری کرده و عمل می نمایند. در واقع هیچ کنترل متمرکزی وجود ندارد و تصمیم گیری ها بصورت محلی انجام میشود. شکل (3) ببینید.

² Sink

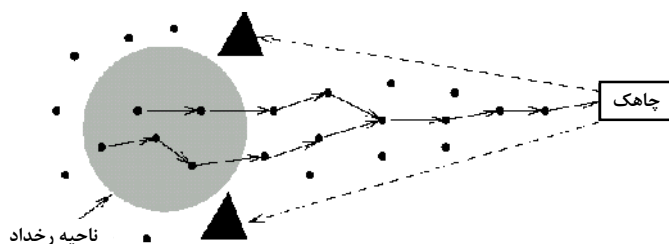
³ Task Manager Node

⁴ Monitoring

ساختار نیمه خودکار: در این ساختار داده ها توسط گره ها به سمت چاهک هدایت شده و فرمان از طریق چاهک به گره های کار انداز ص . شکل (3) را مشاهده کنید



شکل (3) ساختار خودکار



شکل (4) ساختار نیمه خودکار

از طرف دیگر در کاربردهای خاصی ممکن است از ساختار بخش بندی شده یا سلولی استفاده شود که در هر بخش یک ⁵ وجود دارد که داده های گره های دسته خود را به چاهک ارسال می کند. در واقع هر سر دسته مانند یک مدخل ⁶ عمل میکند.

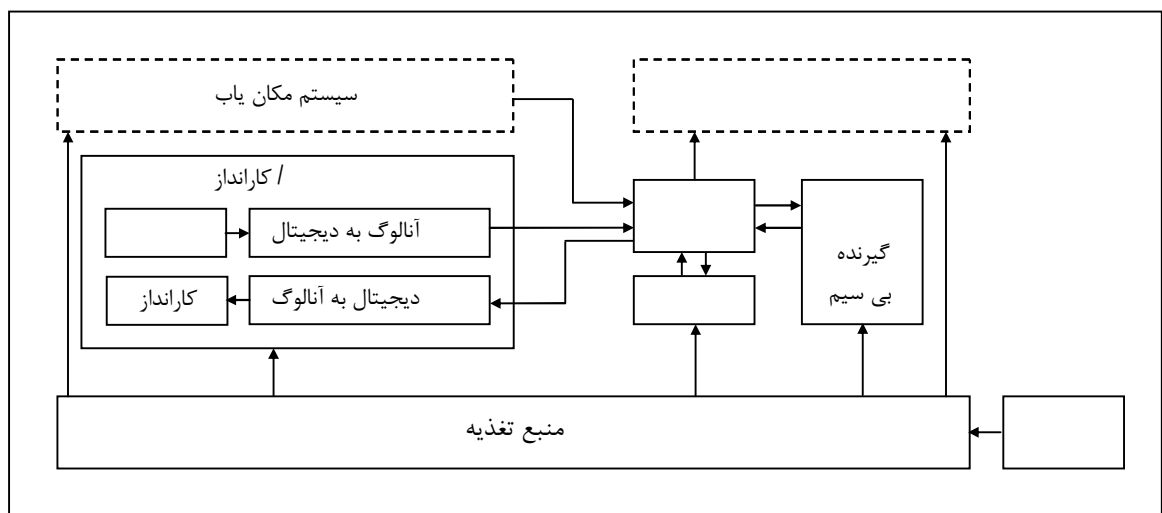
شکل (5) ساختمان داخلی گره حس/کار را نشان می دهد. / کارانداز، / گیرنده بی سیم و منبع تغذیه می باشد بخشهای اضافی واحد متحرک ساز، سیستم مکان یاب و تولید توان نیز ممکن است بسته به کاربرد در گره ها وجود داشته باشد.

شامل یک کوچک و یک با ظرفیت محدود است داده ها را از حسگرها گرفته بسته به کاربرد پردازش محدودی روی آنها انجام داده و از طریق فرستنده ارسال می کند. واحد پردازش مدیریت هماهنگی و

⁵ Cluster_Head

⁶ Gateway

مشارکت با سایر گره ها در شبکه را انجام می دهد. واحد فرستنده گیرنده ارتباط گره با شبکه را برقرار می کند. شامل یک سری حسگر و مبدل آنالوگ به دیجیتال است که اطلاعات آنالوگ را از حسگر گرفته و بصورت دیجیتال به پردازنده تحویل می دهد. واحد کارانداز شامل کارانداز و مبدل دیجیتال به آنالوگ است که فرامین دیجیتال را از پردازنده گرفته و به کارانداز تحویل می دهد. واحد تامین انرژی، توان مصرفی تمام بخشها را تامین می کند که اغلب یک باتری با محدودیت منبع انرژی یکی از تنگناهای اساسی است که در طراحی شبکه های حس/کار همه چیز را تحت تاثیر قرار می دهد. در کنار این بخش ممکن است واحدی برای تولید انرژی مثل سلول های خورشیدی وجود داشته . مکان یاب موقعیت فیزیکی گره را تشخیص می دهد. تکنیکهای مسیرهدهی و وظایف حسگری به اطلاعات مکان با دقت بالا نیاز دارند. یکی از مهمترین مزایای شبکه های /کار توانایی مدیریت ارتباط بین گره های در حال حرکت می باشد.



شکل (5) ساختمان داخلی گره حسگر/کارانداز

ویژگی ها

وجود برخی ویژگی ها در شبکه / کارانداز، آن را از سایر شبکه های سنتی و بی سیم متمایز می کند.

- تنگناهای سخت افزاری شامل محدودیتهای اندازه فیزیکی، ظرفیت حافظه
- تعداد بسیار زیاد گره ها
- چگالی بالا در توزیع گره ها در ناحیه عملیاتی
- داد خرابی در گره ها
- تغییرات توپولوژی بصورت پویا و احيانا متناوب
- استفاده از روش پخش همگانی⁷ در ارتباط بین گره ها در مقابل ارتباط نقطه به نقطه
- بودن شبکه به این معنی که گره ها کد شناسایی⁸

⁷ Broadcast

⁸ Data Centric

⁹ ID (Identification code)

طراحی شبکه های حس/کار موثر است و موضوعات بسیاری در این زمینه مطرح است که بررسی تمام آنها در این نوشتار نمیگنجد از این رو تنها به ذکر برخی از آنها بطور خلاصه اکتفا می کنیم.

1 : هر گره ضمن اینکه باید کل اجزاء لازم را داشته باشد باید بحد کافی کوچک, سبک و کم حجم نیز باشد بعنوان مثال در برخی کاربردها گره باید به کوچکی یک قوطی کبریت باشد و حتی گاهی حجم گره محدود به یک سانتیمتر مکعب است و از نظر وزن آنقدر باید سبک باشد که بتواند همراه باد در هوا معلق شود. در عین حال هر گره باید توان مصرفی بسیار کم, قیمت تمام شده پایین داشته و با شرایط محیطی سازگار باشد. اینها همه محدودیتهایی است که کار طراحی و ساخت گره های حس/کار را با چالش مواجه میکند. ارائه طرح های سخت افزاری سبک و کم حجم در مورد هر یک از اجزای گره بخصوص قسمت ارتباط بی سیم و حسگرها از جمله موضوعات تحقیقاتی است که جای کار بسیار دارد. پیشرفت فن آوری ساخت مدارات مجتمع با فشردگی بالا و مصرف پایین, نقش بسزایی در کاهش تنگناهای

2 : توپولوژی ذاتی شبکه حس/کار توپولوژی گراف است. بدلیل اینکه ارتباط گره ها بی سیم و بصورت پخش همگانی است و هر گره با چند گره دیگر که در محدوده برد آن قرار دارد ارتباط دارد. الگوریتم های کارا در جمع آوری داده و کاربردهای ردگیری اشیاء شبکه را درخت پوشا در نظر می گیرند. چون ترافیک اصولا بفرمی است که داده ها از چند گره به سمت یک گره حرکت می کند. مدیریت توپولوژی باید با دقت انجام شود یک مرحله اساسی مدیریت توپولوژی راه اندازی اولیه شبکه است گره هایی که قبلا هیچ ارتباط اولیه ای نداشته اند در هنگام جایگیری و شروع بکار اولیه باید بتوانند با یکدیگر ارتباط برقرار کنند. الگوریتم های مدیریت توپولوژی در راه اندازی اولیه باید امکان عضویت گره های جدید و حذف گره هایی که بدلایلی از کار می افتند را فراهم کنند. پویایی توپولوژی از خصوصیات شبکه های حس/کار است که امنیت آن را به چالش می کشد. ارائه روشهای مدیریت توپولوژی پویا بطوری که موارد امنیتی را هم پوشش دهد از موضوعاتی است که جای کار زیادی دارد.

3 **قابلیت اطمینان**: هر گره ممکن است خراب شود یا در اثر رویدادهای محیطی مثل تصادف یا انفجار بکلی نابود شود یا در اثر تمام شده منبع انرژی از کار بیفتد. منظور از تحمل پذیری یا قابلیت اطمینان این است که خرابی گره ها نباید عملکرد کلی شبکه را تحت تاثیر قرار دهد. در واقع می خواهیم با استفاده از اجزای غیر قابل اطمینان یک شبکه قابل اطمینان بسازیم. k با نرخ خرابی λ_k قابلیت اطمینان (1) مدل می شود. که در واقع احتمال عدم خرابی t بشرط اینکه گره در بازه زمانی $(0,t)$ خرابی نداشته باشد. به این ترتیب هرچه زمان می خرابی گره بیشتر می

$$(1) \quad R_k(t) = e^{-\lambda_k t}$$

4 **مقیاس پذیری**: شبکه باید هم از نظر تعداد گره و هم از نظر میزان پراکندگی گره ها, مقیاس پذیر باشد. دیگر شبکه حس/کار از طرفی باید بتواند با تعداد صدها, هزارها و حتی میلیون ها گره کار کند و از طرف دیگر, چگالی توزیع متفاوت گره ها را نیز پشتیبانی کند. چگالی طبق فرمول (2) محاسبه می شود. که بیانگر تعداد متوسط گره هایی است که در برد یک گره نوعی (دایره ای با قطر 10) قرار می گیرد. A : مساحت ناحیه کاری N : تعداد گره در ناحیه کاری R : برد ارسال رادیویی است. در بسیاری کاربردها توزیع گره ها اتفاقی صورت می گیرد و امکان توزیع با چگالی مشخص

و یکنواخت وجود ندارد یا گره ها در اثر عوامل محیطی جابجا می شوند. بنابراین چگالی باید بتواند از چند عدد تا چند صد گره تغییر کند. موضوع مقیاس پذیری به روشها نیز مربوط می شود برخی روشها ممکن است مقیاس پذیر نباشد یعنی در یک چگالی یا تعداد محدود از گره کار کند. در مقابل برخی روشها مقیاس پذیر هستند

$$(2) \quad \mu(R) = (N \cdot \pi R^2) / A$$

5 قیمت تمام شده : چون تعداد گره ها زیاد است کاهش قیمت هر تک گره اهمیت زیادی دارد. تعداد گره ها گاهی تا میلیونها میرسد. در این صورت کاهش قیمت گره حتی به مقدار کم تاثیر قابل توجهی در قیمت کل شبکه خواهد داشت.

6 شرایط محیطی : طیف وسیعی از کاربرد های شبکه ه /کار مربوط به محیط هایی می شود که انسان نمی . مانند محیط های آلوده از نظر شیمیایی، میکروبی، هسته ای ویا مطالعات در کف اقیانوس ها و فضا ویا محیط های نظامی بعثت حضور دشمن ویا در جنگل و زیستگاه جانوران که حضور انسان باعث فرار آنها می شود. شرایط محیطی باید در طراحی گره ها در نظر گرفته شود مثلاً در دریا و محیط های مرطوب گره حسگر در محفظه ای که رطوبت را منتقل نکند قرار می گیرد.

7 رسانه ارتباطی: در شبکه های حس/کار ارتباط گره ها بصورت بی سیم و از طریق رسانه رادیویی، یا رسانه های نوری دیگر صورت می گیرد. اکثراً از ارتباط رادیویی استفاده می شود. آسانتر است ولی فقط در خط مستقیم عمل می کند.

8 توان مصرفی : گره های شبکه حس/کار باید توان مصرفی کم داشته باشند. گاهی منبع تغذیه یک باتری 1/2 5/5. آمپر ساعت است که باید توان لازم برای مدت طولانی مثلاً 9 ماه را تامین کند. در بسیاری از کاربردها باتری قابل تعویض نیست. لذا عمر باطری عملاً عمر گره را مشخص می کند. بعثت اینکه یک گره علاوه بر () یا اجرای یک فرمان (توسط کارانداز) بعنوان رهیاب¹⁰ نیز عمل می کند بد عمل کردن گره باعث حذف آن از توپولوژی شده و سازماندهی مجدد شبکه و مسیردهی مجدد بسته عبوری را در پی خواهد داشت. طراحی سخت افزار گره ها استفاده از طرح ها و قطعاتی که مصرف پایینی دارند و فراهم کردن امکان حالت خواب¹¹ کل گره یا برای هر بخش بطور مجزا مهم است.

9 افزایش طول عمر شبکه: یک مشکل این است که عمر شبکه های حس/کار نوعاً کوتاه است. گره ها بعثت محدودیت انرژی منبع تغذیه کوتاه است. علاوه بر آن گاهی موقعیت ویژه یک گره در شبکه مشکل را تشدید می کند مثلاً در گره ای که در فاصل یک قدمی چاهک قرار دارد از یکطرف بخاطر بار کاری زیاد خیلی زود انرژی خود را از دست می دهد و از طرفی از کار افتادن آن باعث قطع ارتباط چاهک با کل شبکه می شود و از کار افتادن شبکه می شود. برخی راه حل ها به ساختار برمی گردد مثلاً در مورد مشکل فوق استفاده از ساختار خودکار راهکار مؤثری است. (2

(بعثت اینکه در ساختار خودکار بیشتر تصمیم گیری ها بطوری محلی انجام می شود ترافیک انتقال از طریق

¹⁰ Router

¹¹ Sleep

گره بحرانی کم شده، طول عمر آن و در نتیجه طول عمر شبکه افزایش می یابد. مشکل تخلیه زود هنگام انرژی در مورد گره های نواحی کم تراکم در توزیع غیر یکنواخت گره ها نیز صدق می کند (4 مراجعه کنید) در اینگونه موارد داشتن یک مدیریت توان در داخل گره ها و ارائه راه حل های توان آگاه بطوری که از گره های بحرانی کمترین استفاده را بکند. این نوعی به اشتراک گذاری منابع محسوب می شود لذا در صورت داشتن مدیریت وظیفه و مدیریت توان مناسب توزیع با چگالی زیاد گره ها در میدان حسگر/ کارانداز طول عمر شبکه را افزایش میدهد. ساختاری مناسب و ارائه روشهای مدیریتی و الگوریتم ها تو با هدف افزایش طول عمر شبکه حس/کار از مباحث تحقیقاتی است.

40 **12 و هماهنگی**¹³ : در برخی کاربردها مانند سیستم تشخیص و جلوگیری از گسترش آتش سوزی یا سیستم پیش گیری از سرقت سرعت پاسخگویی شبکه اهمیت زیادی دارد. در نمایش بلادرنگ فشار بر رو مانیتور بسته های ارسالی باید بطور لحظه ای روزآمد باشند. برای تحقق بلادرنگ یک روش این است که برای بسته های ارسالی یک ضرب العجل تعیین شود و در لایه کنترل دسترسی رسانه¹⁴ بسته های با ضرب العجل کوتاهتر زودتر ارسال شوند مدت ضرب العجل به کاربرد بستگی دارد. لیه مهم دیگر تحویل گزارش رخدادها به چاهک، یا کارانداز ناحیه، ترتیب وقوع آنهاست در غیر این صورت ممکن است شبکه واکنش درستی انجام ندهد. نکته دیگر هماهنگی کلی شبکه در ارتباط با گزارشهایی است که در مورد یک رخداد از حسگرهای مختلف به کاراندازهای ناحیه مربوطه داده می شود. مثال در یک کاربرد نظامی فرض کنید حسگرهایی جهت تشخیص حضور یگان های پیاده دشمن و کاراندازهایی جهت نابودی آن در نظر گرفته شده چند حسگر حضور دشمن را به کار اندازها اطلاع می دهند شبکه باید در کل منطقه، عملیات را به یکباره شروع کند. در غیر این صورت با واکنش اولین کارانداز، سربازان دشمن متفرق شده و عملیات با شکست مواجه می شود. بهرحال موضوع بلادرنگ و هماهنگی در شبکه های حس/کاربخصوص در مقیاس بزرگ و شرایط نامطمئن از مباحث تحقیقاتی است.

41 **امنیت**¹⁵ : موضوع امنیت در برخی کاربردها بخصوص کاربرد های نظامی یک موضوع بحرانی است و بخاطر برخی ویژگی ها شبکه های حس/کار در مقابل مداخلات آسیب پذیر ترند. یک مورد بی سیم بودن ارتباط شبکه است که کار دشمن را برای فعالیت های ضد امنیتی و مداخلات آسانتر می کند. مورد دیگر استفاده از یک فرکانس واحد ارتباطی برای کل شبکه است که شبکه را در مقابل استراق سمع آسیب پذیر می کند. مورد بعدی ویژگی پویایی توپولوژی است که زمینه را برای پذیرش گره های دشمن فراهم می کند. اینکه پروتکل های مربوط به مسیردهی، کنترل ترافیک و لایه کنترل دسترسی شبکه سعی دارند با هزینه و سربار¹⁷ کمتری کار کنند مشکلات امنیتی بوجود می آورد مثلا برای شبکه های حسگر در مقیاس بزرگ برای کاهش تأخیر بسته هایی که در مسیر طولانی در طول شبکه حرکت می کنند یک راه حل خوب این است که اولویت مسیردهی به بسته های عبوری داده شود. همین روش باعث می شود حمله های سیلی¹⁸ یکی از نقاط ضعف شبکه حس/کار کمبود منبع انرژی است و دشمن می تواند با قرار دادن یک گره مزاحم که مرتب پیغام های بیدار باش بصورت پخش همگانی با انرژی زیاد تولید می کند باعث شود بدون دلیل گره

¹² Real-time

¹³ Coordination

¹⁴ MAC (Medium Access Control)

¹⁵ Security

¹⁶ Interferences

¹⁷ Overhead

¹⁸ Flooding Attack

های همسایه از حالت خواب¹⁹ . ادامه این روند باعث به هدر رفتن انرژی گره ها شده و عمر آنها را کوتاه می کند. با توجه به محدودیت ها باید دنبال راه حل های ساده و کارا مبتنی بر طبیعت شبکه حس/کار . مثلا اینکه گره ها با چگالی بالا می توانند توزیع شوند و هر گره دارای اطلاعات کمی است یا اینکه داده ها در یک مدت کوتاه معتبرند از این ویژگی ها می توان بعنوان یک نقطه قوت در رفع مشکلات امنیتی استفاده کرد. چالشهای زیادی در مقابل امنیت شبکه حس/کار وجود دارد. و مباحث تحقیقاتی مطرح در این زمینه گسترده و پیچیده است.

12 عوامل پیش بینی نشده: یک شبکه حسگر کارانداز تابع تعداد زیادی از عدم قطعیت هاست. عوامل طبیعی غیر قابل پیش بینی مثل سیل زلزله، مشکلات ناشی از ارتباط بی سیم و اختلالات رادیویی، امکان خرابی هر گره، کالیبره نبودن، پویایی ساختار و مسیردهی شبکه، اضافه شدن گره های جدید و حذف گره های قدیمی، جابجایی گره ها بطور کنترل شده یا در اثر عوامل طبیعی و غیره. سوآلی که مطرح است این است که در این شرایط چگونه میتوان چشم اندازی فراهم کرد که از دیدگاه لایه کاربرد شبکه یک موجودیت قابل اطمینان در مقیاس بزرگ دارای کارایی عملیاتی مشخص و باتوجه به اینکه شبکه های حسگر کارانداز تا حدود زیادی بصورت مرکزی غیر قابل کنترل هستند و بصورت خودکار یا حداقل نیمه خودکار عمل میکنند باید بتوانند با مدیریت مستقل بر مشکلات غلبه کنند. از این رو باید ویژگی های خود بهینه سازی²⁰ خود سازماندهی²¹ و خود درمانی²² . اینها از جمله مواردی هستند که بحث در مورد آنها آسان ولی تحقق آن بسیار پیچیده است. بهر حال این موضوعات از جمله موارد تحقیقاتی می باشند

نمونه ی پیاده سازی شده شبکه / کار

ذره ی میکا²³

یک نمونه از پیاده سازی سخت افزاری گره های حسگر ذره میکا دانشگاه برکلی امریکا است. این ، یک واحد /کار کوچک (چندین اینچ مکعب) با یک واحد پردازنده مرکزی²⁴، منبع تغذیه، رادیو و چندین عنصر حسگر اختیاری می . پردازشگر آن یک پردازنده 8 بیتی از خانواده ی اتمل²⁵ می باشد همراه با 128 کیلو بایت حافظه ی برنامه، 4 کیلوبایت RAM 512 کیلوبایت حافظه ی فلش. این پردازنده فقط یک کمینه از مجموعه دستورالعمل های ریسک (RISK) ، شیفت با طول متغیر و چرخش پشتیبانی می کند. رادیوی آن یک رادیوی مصرف پایین 916 کیلو در ثانیه روی یک کانال تسهیم شده منفرد با محدوده ی نزدیک 12 متر می . رادیو در حالت دریافت 4.8 میلی آمپر، 12 میلی آمپر و در حالت خواب 5 میکرو آمپر مصرف می کند.

¹⁹ Sleep

²⁰ Self Optimizing

²¹ Self Organizing

²² Self Healing

²³ MICA MOTE

²⁴ CPU

²⁵ 8-bit 4 MHz Atmel ATmega 128



شکل (6) ذره میکا

ذره میکا

کوچکترین آن اغلب به عنوان ²⁶ شناخته می شود. پژوهشی غبار هوشمند که به وسیله ی پروفیسور پیتستر ²⁷ و کان ²⁸ رهبری و هدایت می شود موفق به دستیابی حدی برای کاهش اندازه برای ساختن گره های ارزان و البته تسهیل گسترش آن بسیار مهم است. گروه تحقیقاتی امیدوارند که ضمن حفظ موثر توانایی های حسگری و ارتباطی می توانند موارد ، مخابره اطلاعات و محاسبات سخت افزاری همراه با منبع تغذیه را در اندازه ای در حدود چند میلیمتر مکعب فراهم کنند. این گره میلیمتر مکعبی غبار هوشمند نام دارد که حقیقتاً قلمرو چیزهای ممکن شدنی است. چنان که نمونه های آتی آن می تواند به قدری کوچک باشد که معلق در هوا باقی مانده و به وسیله جریان هوا شناور شود و برای ساعت ها یا روزها موارد حس شده را ارسال کند. غبار هوشمند می تواند اطلاعات را با استفاده از یک تکنولوژی بازتابنده ی نوری جدید، به صورت غیر فعال ³⁰ ارسال کند این یک راه معقول و ارزان برای پراب ³¹ یک سنسور یا تایید دریافت اطلاعات را فراهم می کند ارسال نوری فعال ³² نیز ممکن است اما اتلاف انرژی بیشتری دارد.

²⁶ Smart Dust

²⁷ Pitster

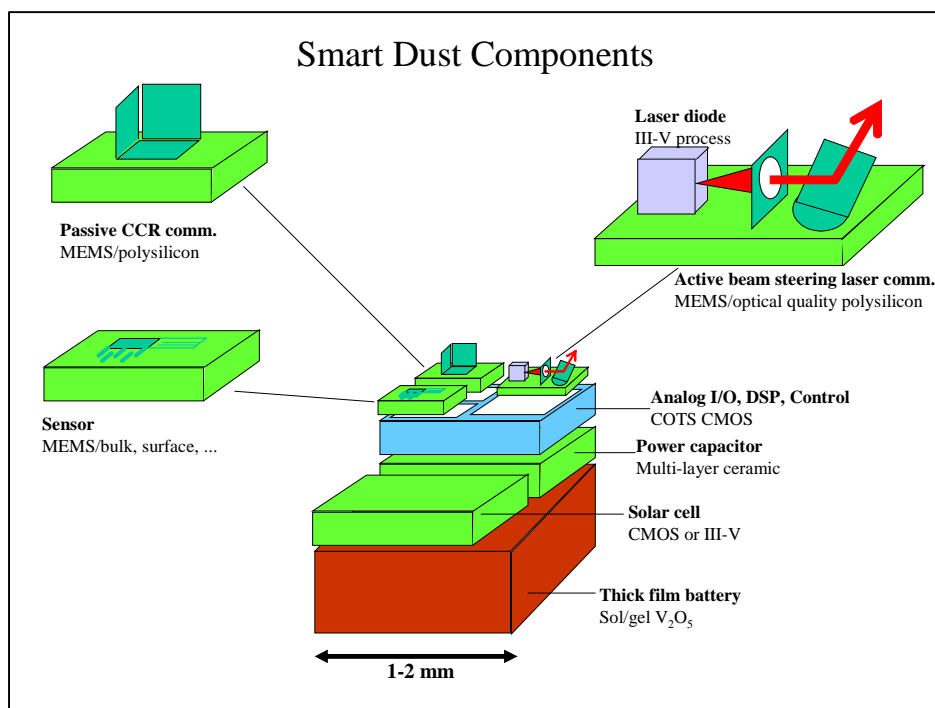
²⁸ Kahn

²⁹ power consumption

³⁰ passively

³¹ probe

³² active



شکل (7) ساختار داخلی غبار هوشمند

بررسی نرم افزارهای شبیه سازی شبکه

امروزه تکنولوژی شبیه سازی به طرز موفقیت آمیزی در جهت مدل سازی، طراحی و مدیریت انواع سیستم های هوشمند به کار گرفته شده و در این راستا ابزارها و تکنیک های متعددی خلق شده که به طور مثال می توان به تکنیک شبیه سازی رویدادگردان اشاره کرد که اساس عملکرد بسیاری از شبیه سازهای نوین می باشد. کاربرد شبیه سازی در مورد شبکه های ارتباطی نیز سابقه ای 15 ساله دارد که هنوز هم در حال رشد می باشد، دلایل استفاده از شبیه سازی در این حوزه را می توان در دو مورد خلاصه کرد:

1 پیدایش و گسترش شبکه هایی باتکنولوژی پیچیده

2 خلق ابزارها و نرم افزارهای خاص شبیه سازی شبکه ها

نرم افزارهای شبیه ساز شبکه توانایی شبیه سازی شبکه های ارتباطی را بدون نیاز به کد نویسی و معمولاً از طریق واسط های گرافیکی فراهم می کنند. وجود عناصر شبیه سازی شده ای متناظر با عناصر واقعی (روتورها و سوئیچ ها، ...) در این گونه موارد علاوه بر بالا بردن دقت، باعث افزایش سهولت و سرعت در فرآیند شبیه سازی می شود و به این ترتیب برای کاربران ناآشنا با فن برنامه نویسی بسیار مناسب می باشد.

خصوصیات لازم برای شبیه سازهای شبکه

خصوصیاتی که شبیه سازهای شبکه باید داشته باشند عبارتند از:

4

کاربر باید قادر باشد انواع جدیدی از منابع معمول شبکه همچون گره ها، لینک ها و پروتکل ها ر شبیه ساز بیفزاید.

وجود واسطه گرافیکی و امکان مدل‌سازی به صورت ساخت یافته، به شکلی که مدل‌های پیچیده بر اساس مدل‌های ساده طرح شوند و همچنین قابلیت استفاده مجدد از ماژول‌ها از خصوصیتی می‌باشد که باعث تسریع در فرآیند شبیه‌سازی می‌شود.

3 اجرای سریع مدل‌ها :

زمان پردازش در شبیه‌سازی‌های بزرگ برای شبکه‌هایی با تعداد زیاد گره بسیار مهم می‌باشد که لازمه آن مدیریت صحیح حافظه می‌باشد.

4 قابلیت مصور سازی :

نمایش گرافیکی عناصر شبکه در حال تبادل پیغام‌ها با یکدیگر به رفع خطاهای شبیه‌سازی و درک نحوه کارکرد آن بسیار کمک می‌کند. در برخی نرم‌افزارهای شبیه‌سازی اجرا می‌شود همزمان با اجرای شبیه‌سازی و در برخی دیگر پس از Play Back انجام می‌گیرد.

5 قابلیت اجرای مجدد و تکراری شبیه‌سازی :

هدف از انجام شبیه‌سازی به طور عمده تحقیق تأثیر یک یا چند پارامتر (برای مثال متوسط طول بسته‌ها و یا ظرفیت) بر کارایی شبکه می‌باشد و به همین خاطر تکرار پذیری یک شرط لازم برای این نرم‌افزارها می‌باشد. باید توجه داشت که خلق یک شبیه‌سازی شبکه دقیق و معتبر مستلزم بکارگیری تکنولوژی شبیه‌سازی در کنار دانش شبکه و پروتکل‌های آن می‌باشد. البته در کنار خصوصیات فوق وجود برخی قابلیت‌ها بر ارزش هر ابزار شبیه‌سازی خواهد افزود که از آن میان می‌توان به چند مورد اشاره ذیل اشاره کرد :

1 وجود ماژول‌های درونی از پیش آماده شده متناظر با عناصر و پروتکل‌های شبکه .
2 وجود یک مولد عدد تصادفی و در شکل‌های پیشرفته‌تر قابلیت خلق کمیت‌های با توزیع‌های تصادفی گوناگون چرا که اغلب رخدادها در یک فرآیند شبیه‌سازی اعم از تولید و ارسال بسته‌ها و یا ایجاد خرابی در آنها، از نوع فرآیندهای تصادفی می‌باشند.

3 حمایت از کاربران به بهنگام سازی‌های به موقع (بخصوص در مورد پروتکل‌های جدید) به‌همراه مستندات کامل و گویا.
4 ارائه گزارش‌هایی از پارامترهای کارایی شبکه (نرخ خروجی، بهره‌وری، تأخیر انتقال،...) در قالب ارقام و منحنی‌ها به همراه امکان انجام عملیات آماری روی نتایج از دیگر ویژگی‌های مثبت یک شبیه‌سازی می‌باشد.

شبیه‌سازی NS(v2)

شروع به کار این نرم‌افزار به پیش از پروژه VINT مربوط می‌گردد. شبیه‌سازی NS 1989 توسط گروه تحقیقاتی شبکه NRG در آزمایشگاه LBNL و بر اساس شبیه‌سازی شبکه دیگری موسوم به REAL طراحی شده است که توسعه آن تا امروز ادامه داشته و بخصوص پس از انتخاب شدن به عنوان ابزار شبیه‌سازی پروژه VINT جدیت و سرعت یافته است. آخرین نسخه آن NS 2.1b8 روی شبکه اینترنت قابل دسترسی می‌باشد.

معماری درونی NS

NS2 از گونه شبیه سازهای رویدادگردان می باشد و از طریق پیگیری رخدادها در طول زمان های گسسته ، شبیه سازی را پیش می برد این شبیه ساز در دو محیط برنامه نویسی ++C و OTCL و بصورت شیئی گرا طرح شده .NS مدلی موسوم به VuSystem کار می کند که در ادامه به طور خلاصه در مورد آن توضیح می دهیم.

VuSystem

شبیه سازهای شبکه عمدتاً از دو بخش با تمایلات گوناگون تشکیل یافته اند :

1 گروهی از بلاک های سازنده که عناصری همچون گره ها ، لینک ها ، صف ها ، مولدهای ترافیک و پروتکل ها را شبیه سازی می کنند.

2 یک رابط که معمولاً از آن تحت عنوان زبان تشریح شبیه سازی یا SDL یاد می شود و وظیفه دارد فوق را در فرآیند شبیه سازی به یکدیگر متصل کند.

در مورد این دو بخش یک مشکل اساسی پیش روی طراحان شبیه سازهای شبکه وجود دارد. در حالی که برای بلوک های سازنده کارایی و سرعت اجرا هدف اصلی می باشد. SDL نیازمند انعطاف و سهولت تغییر در پیکر بندی می باشد و نیل به این دو هدف با یک محیط برنامه نویسی واحد مشکل مینماید. بنابراین مدل VuSystem که توسط David Wetherall MIT پیشنهاد شده ، راه حل را در استفاده از دو محیط برنامه نویسی جداگانه برای دو بخش فوق می داند. مطابق این مدل بلوک های سازنده با یک زبان کامپایلی (++C) و بخش رابط آنها در یک محیط (OTCL) پیاده سازی می شوند.

NS-2 با بکار گیری مجموعه ای از اشیاء موسوم به اشیاء دو تکه موفق به اعمال مدل VuSystem در شبیه ساز خویش گشته اند . مطابق این مدل NS-2 متشکل از مجموعه ای از اشیاء می باشد که در دو محیط دوگانه کامپایلی / مفسری و از طریق فراخوانی متدهای یکدیگر ، ارتباط برقرار می کنند.

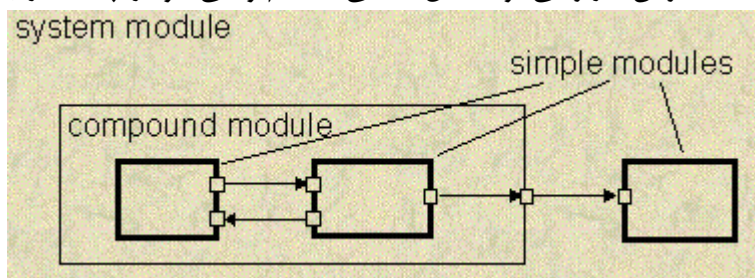
شبیه ساز OMNeT++

OMNeT++ یک شبیه ساز شیئی گرا می باشد و از دسته نرم افزارهای discrete event . OMNeT++ Objective Modular Network است و مبتنی بر ++C . چون این نرم افزار در محیط ++C شده است در اغلب محیط ها با کامپایلر ++C قابل اجرا می باشد. به وسیله DoS - window پشتیبانی شده و WinNT Win95 Win3.1 . نویسنده این نرم افزار با یک شرکت مجارستانی توزیع کننده OPNET همکاری داشته است. این فرد عضو چندین پروژه شبیه سازی شبکه بوده و مدل شبیه سازی سیستم VSAT OPNET . از لحاظ ساختاری این نرم افزار سلسله مراتبی از ماژول های تو در تو می باشد که ماژول ها از طریق تبادل پیغام با یکدیگر در ارتباط هستند . ائین ترین سطح این سلسله مراتب ماژول های خود کاربر قابل ایجاد می باشند. می تواند اجرای شبیه سازی را به صورت موازی پیش ببرد. با این نرم افزار هر نوع مکانیزم زمانبندی قابل بکار گیری است. زبان متنی برای توصیف توپولوژی دارد که به آن NED گویند و بوسیله هر ابزار پردازشگر متنی (perl)

awk) قابل ایجاد است. همین فرمت بوسیله ادیتور گرافیکی قابل بکار گیری است. OMNeT++

دییگ کردن و تریس کردن فراهم می کند.

همان طور که ذکر شد OMNeT++ ساختار ماژولی دارد بنابراین تمام ابزار مورد نیاز در قالب ماژول هستند. این ساختار سلسله مراتبی دارند. در بالاترین سطح ماژول سیستم قرار دارد. ماژول سیستم حاوی زیر ماژول ها می باشد. که این زیر ماژول ها می توانند حاوی زیر ماژول های دیگری باشند الی آخر. شمایی از ساختار ماژولی در شکل آمده است. ماژول های تو در تو محدود نیست. بنابراین کاربر می تواند مدل منطقی سیستم واقعی خود را پیاده سازی کند.



شکل (8)

دو نوع ماژول داریم ماژول های مرکب و ساده. ماژولهای مرکب همانطور که از نامش پیداست به ماژول هایی گفته می شود که زیر ماژول ها را در بر دارند. ماژول های ساده ماژول هایی هستند که بوسیله خود کاربر ایجاد می شوند. در حقیقت ماژول های ساده الگوریتم های مدل هستند.

همه ماژول های سیستم در قالبی به نام module type . کاربر برای توصیف مدل modul type ها را به کار می گیرد (نمونه هایی از module type را برای ایجاد module type های پیچیده تر بکار می گیرد).

ماژول سیستم نمونه ای از module type های از پیش تعریف شده است. زمانی که module type سازنده بکار گرفته شود تفاوتی بین ماژول ساده و ماژول مرکب نیست. بدین مفهوم که کاربر برای سادگی می تواند یک ماژول ساده را به چندین ماژول ساده بشکند و در قالب یک ماژول مرکب بگنجانند یا برعکس قابلیت های یک ماژول مرکب را در یک ماژول ساده خلاصه کند.

همان طور که ذکر شد ماژول ها از طریق تبادل پیغام با یکدیگر ارتباط برقرار می کنند. در شبکه واقعی پیام ها می توانند فریم ها یا بسته ها باشند. ه از طریق ارسال مستقیم پیغام یا به کمک مسیره های از پیش تعریف شده با

یکدیگر در ارتباط هستند. گیت ها واسط های ماژول ها هستند که دارای بافر می باشند و عامل اتصال دهنده لینک ها به یکدیگرند. لینک ها تنها در یک سطح از سلسله مراتب ماژول قابل ایجاد هستند. بدین معنا که در یک ماژول مرکب دو زیر ماژول می توانند از طریق گیت های متناظر متصل شوند و یا یک زیر ماژول با ماژول مرکب خود مرتبط شود.

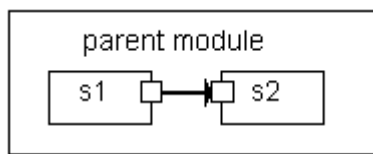
در ساختار سلسله مراتبی پیام ها از طریق لینک ها یا اتصالات قابل انتقال هستند که مبدأ و مقصد پیغام ها ماژول های ساده می باشند. به سری لینک ها یا اتصالاتی که از یک ماژول ساده شروع و به یک ماژول ساده ختم می شود مسیر (route) گویند.

پارامترهای زیر را می توان برای یک لینک مقداردهی نمود:

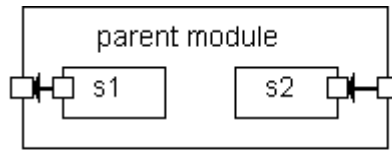
Propagation Delay (sec) 1

bit error rate (errors/bit) 2

data rate (bits/sec) 3



Submodules connected to each other



Submodules connected to the parent module

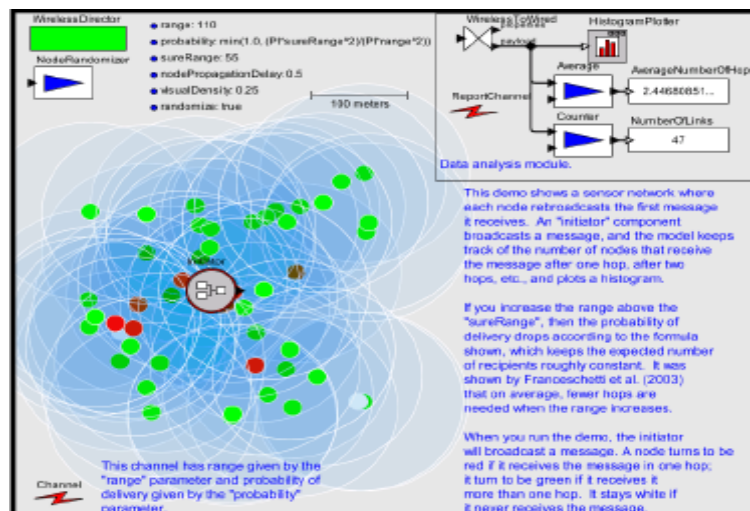
شکل (9)

شبیه ساز Ptolemy II

این گزارش مدل سازی و چهارچوب شبیه سازی نرم افزاری را که visualsense نامیده می شود و برای شبکه های ³³ به کار گرفته می شود را شرح می دهد. این نرم افزار بر روی نرم افزار Ptolemy II (بطلمیوس) تشکیل شده

این چهارچوب تعریف اکتور گرا³⁴ از شبکه های حسگر، کانال های انتقال بی سیم، رسانه های فیزیکی از قبیل کانال صوتی و زیر سیستم های سیمی³⁵ را پشتیبانی می کند.

ساختار نرم افزار شامل یک مجموعه ای از کلاس های پایه برای تعریف کانال ها و گره های حسگر است. یک کتابخانه از زیر کلاس ها، برخی مدل های کانال و مدل های گره و یک چهارچوب ویروالی گسترش پذیر را، فراهم می کند. های مرسوم می توانند بوسیله زیر کلاس کلاسهای پایه و تعریف رفتار در جاوا یا بوسیله ایجاد مدل های ترکیبی با استفاده از چند محیط مدل سازی Ptolemy II ایجاد و تعریف گردند.



شکل (10)

³³ Sensor network

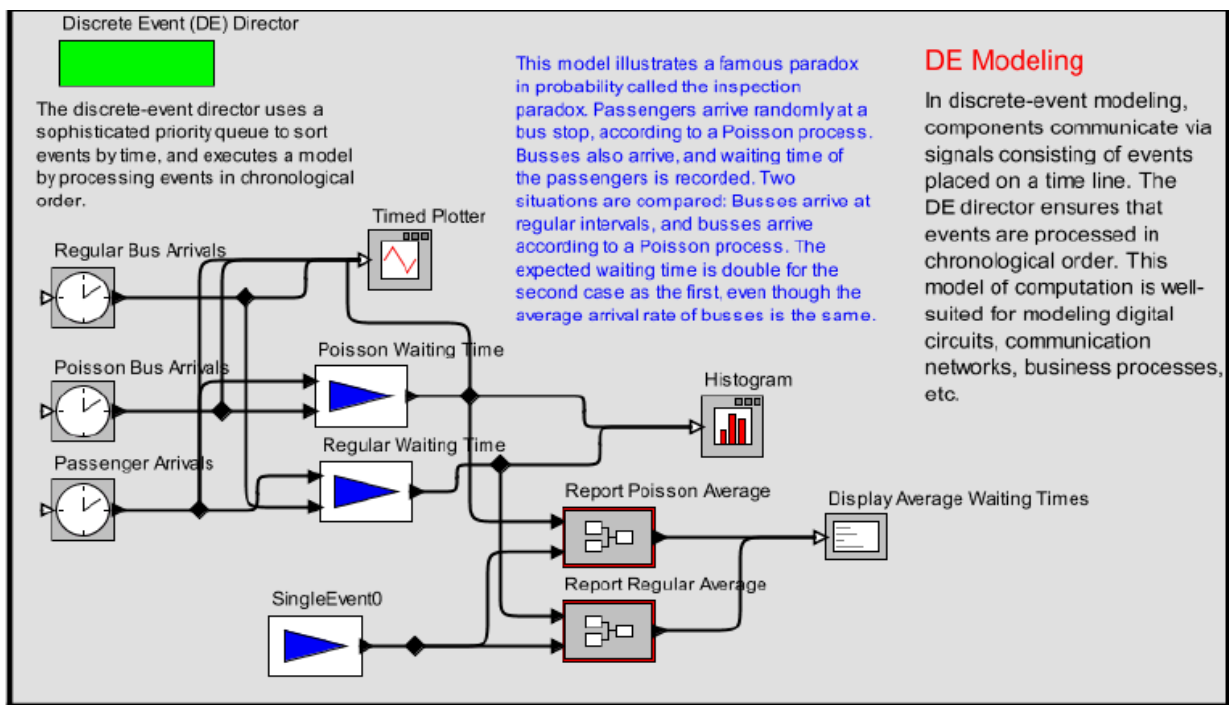
³⁴ Actor-oriented

³⁵ Wired

VisualseNSE یک پیکر بندی از Ptolemy II است که طراحی مؤلفه گرا از شبکه های حسگر و شبیه سازی طبیعی و ویژوالی آن را فراهم می کند.

DE³⁶ Ptolemy II مدل های با توپولوژی های اتصال داخلی که به طور پویا تغییر می کنند، را پشتیبانی می کند. تغییرات در اتصالات به عنوان تغییر در ساختار مدل بحث می شود. نرم افزار به طور دقیق برای پشتیبانی از دسترسی چند نخ کشی شده³⁷ به این قابلیت تغییر و دگرگونی طراحی شده است. بنابراین یک نخ می تواند شبیه سازی مدل را در حالی اجرا کند که نخ دیگر ساختار مدل را تغییر می دهد، برای مثال بوسیله اضافه کردن، حذف یا حرکت اکتورها تا تغییرات اتصالات میان اکتورها، نتیجه قابل پیشگویی و سازگار است.

سر راست ترین استفاده های دامنه DE Ptolemy II شبیه به دیگر چهارچوب های مدل کننده رویداد گسسته از قبیل VHDL, OPNET, NS (که اکتور نامیده می شوند) ورت هایی دارند و پورت ها به هم وصل می شوند تا توپولوژی انتقال را مدل کند. Ptolemy II یک ویرایشگر ویژوالی³⁸ برای ساخت به صورت بلوک دیاگرام تهیه می کند. (شکل 11)



شکل (11) DE Ptolemy, به عنوان بلوک دیاگرام نمایش داده شده است.

مدل سازی شبکه های بی سیم:

در این بخش، ما شرح می دهیم که چطور مدل هایی از شبکه های حسگر بی سیم³⁹ ایجاد کنید و آن را اجرا کنید.

³⁶ Discrete event

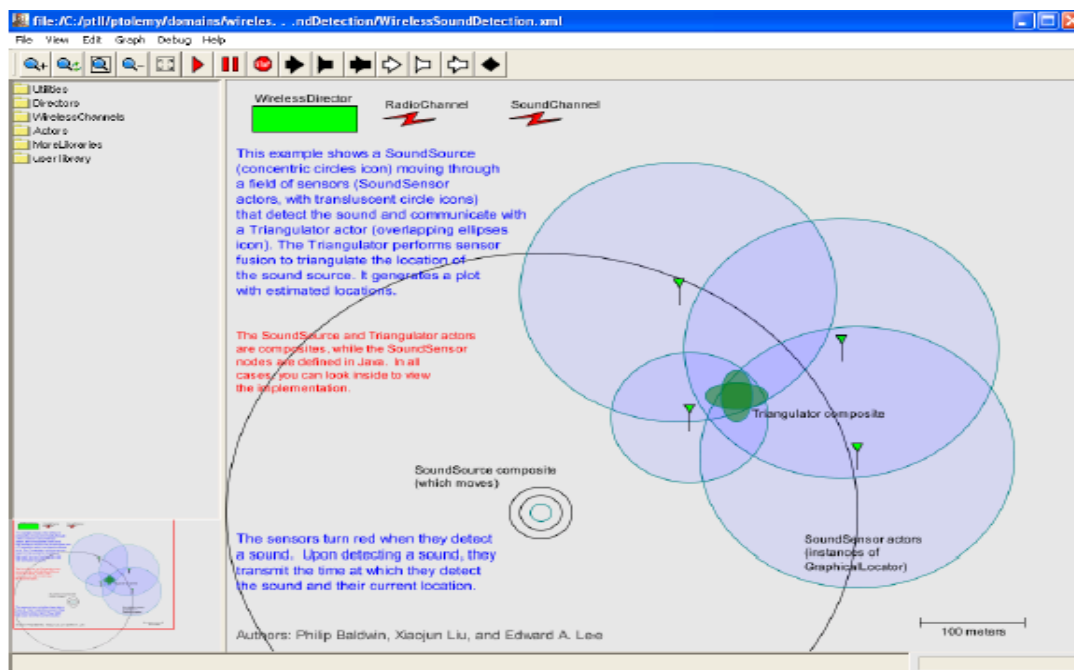
Multithreaded³⁷

³⁸ Visual Editor

³⁹ Wireless Sensor Network

اجرای یک مدل پیش ساخته:

این یک مدل ساده شده از یک سیستم متمرکز صوتی است که در آن از میدانی از گره های حسگر استفاده می کند که یک صدا را شناسایی و بوسیله پیام رادیویی به مرکز⁴⁰ گزارش می کند تا موقعیت صدا را مثلث بندی⁴¹ کند. شکل 12 می دهد که مدل شامل Wireless Director⁴²، که به عنوان یک مدل بی سیم تعریف می کند، دو مدل کانال (یک مدل کانال رادیویی و یک مدل کانال صوتی) یک تعدادی یادداشت⁴³ (و اکتور هایی در مدل است. هر یک از این مؤلفه ها یک نقش در مدل ایفا می کند. هدایت کننده⁴² در اجرای مدل پا به میان می گذارد. مدل های کانال با ارتباط میان اکتور ها سروکار دارد. اکتور ها سیگنال هایی از طریق کانال می فرستند و دریافت می کنند.



شکل (12) نمایش Visualese wireless sound detection

روی مثلث قرمز رنگ در نوار ابزار کلیک کنید، در نتیجه اکتور منبع صوت (که با دواپس متحد المركز شفاف نمایش داده شده است) در یک الگوی دایره ای شروع به حرکت می کند که بوسیله یک فلش آبی رنگ در شکل 13 پش داده شده است. اکتور منبع صوت رویداد هایی از طریق مدل کانال صوتی منتشر می کند. این رویدادها با یک تاخیر زمانی که بستگی به فاصله میان گره های دایره ای آبی رنگ دارد، منتشر می شود. موقعی که این گره ها صدا را شناسایی می کنند، آنها یک سیگنال رادیویی از طریق مدل کانال رادیویی پخش می کنند و آیکون های آنها به رنگ قرمز تغییر می کند تا به طور ویژوالی نشان دهد که آنها این کار را انجام داده اند. سیگنال های رادیویی شامل یک مهر زمانی رویداد صوتی شناسایی شده است. اکتور مثلثی⁴³ در مرکز (که با یک آیکون سبز رنگ نشان داده است) این سیگنال رادیویی را

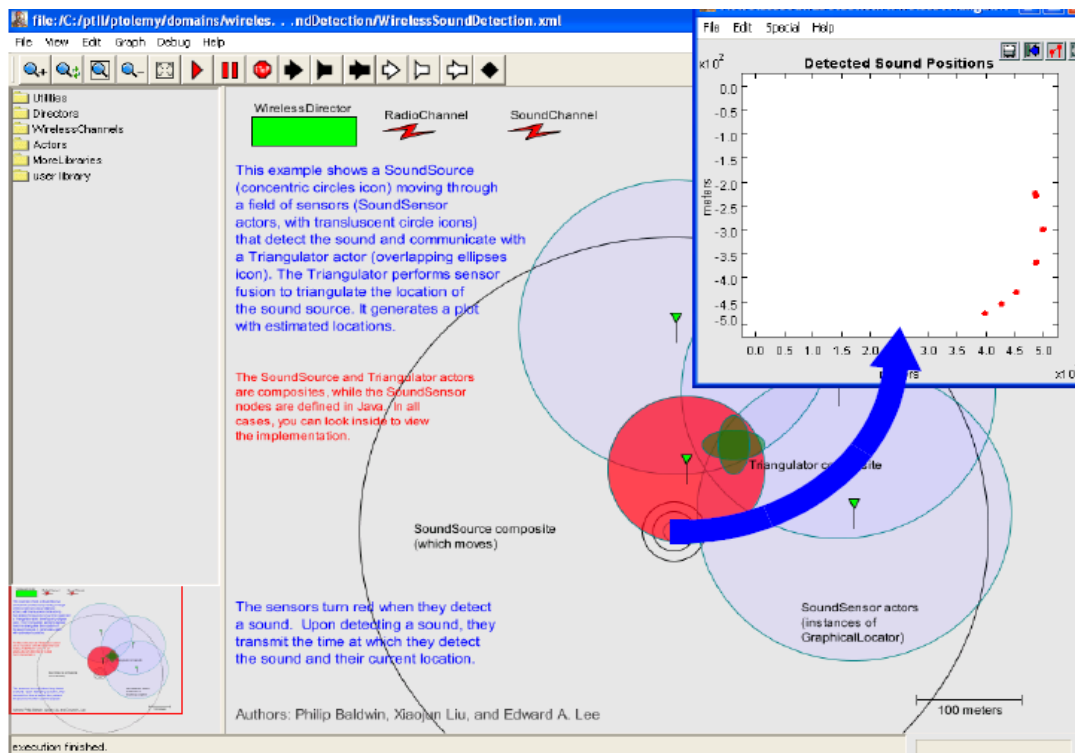
⁴⁰ Hub

⁴¹ Triangulate

⁴² Director

⁴³ Triangulator

دریافت می کند) و مهر های زمانی را برای تخمین موقعیت منبع صوت استفاده می کند. سپس موقعیت را ترسیم می کند، نتیجه در نمودار شکل 13



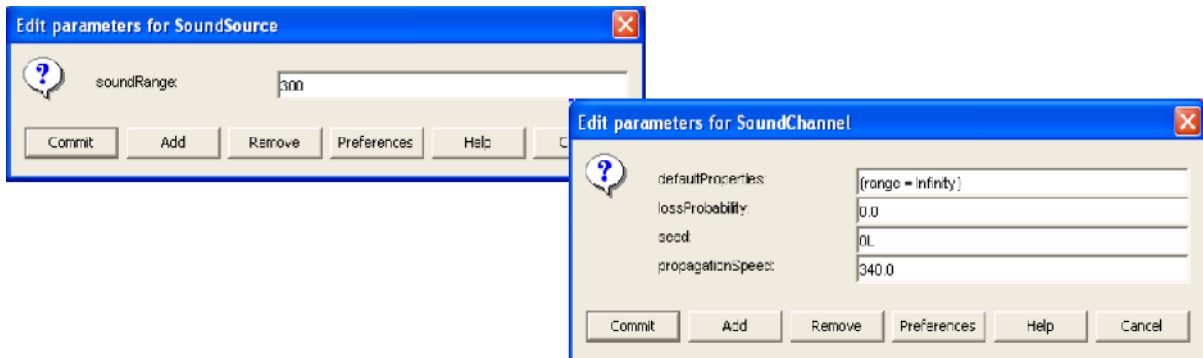
شکل (13) نمایش مدل در حال اجرا

تغییر پارامترها :

ارامترهایی دارد که شما می توانید با آنها آزمایش انجام دهید. پارامترهای دو مؤلفه ی منبع صوت و کانال صوتی در شکل 14 برای بدست آوردن صفحه پارامترها روی اکتور دابل کلیک کنید و یا روی آن راست کلیک

کرده و گزینه *configure* را انتخاب کنید. این منبع صوت یک پارامتر دارد که *soundRange* 300 500 متر آنگاه آیگون دایره ای برای اکتور افزایش می یابد و اجرای مجدد مدل منجر به افزایش خط سیر منبع صوت مثلث بندی شده⁴⁴ می شود. در پارامترهای کانال صوتی، شما می توانید یک مقدار غیر صفر برای *lossProbability* تنظیم کنید، در این مورد فقط بعضی از رویداد های صوتی شناسایی خواهند شد. تنظیم *seed* یک مقدار غیر صفر باعث می شود تا آزمایش تکرار پذیر باشد، بدین معنی که هر آزمایش همان توالی از اعداد را ایجاد خواهد کرد. رها کردن *Seed* مقدار پیش فرض "OL" آزمایش جدیدی را در هر اجرا ثمر می دهد.

⁴⁴ Triangulated



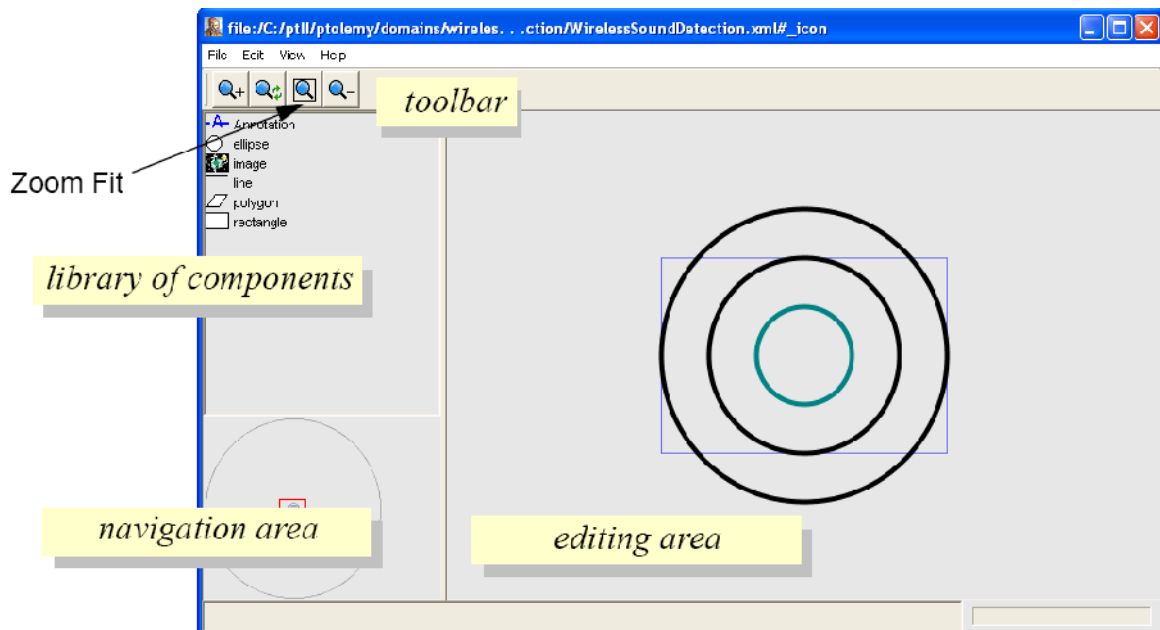
شکل (14) پارامتر های اکتور منبع صوت () و مدل کانال صوتی ()

ساختار یک مدل پیش ساخته :

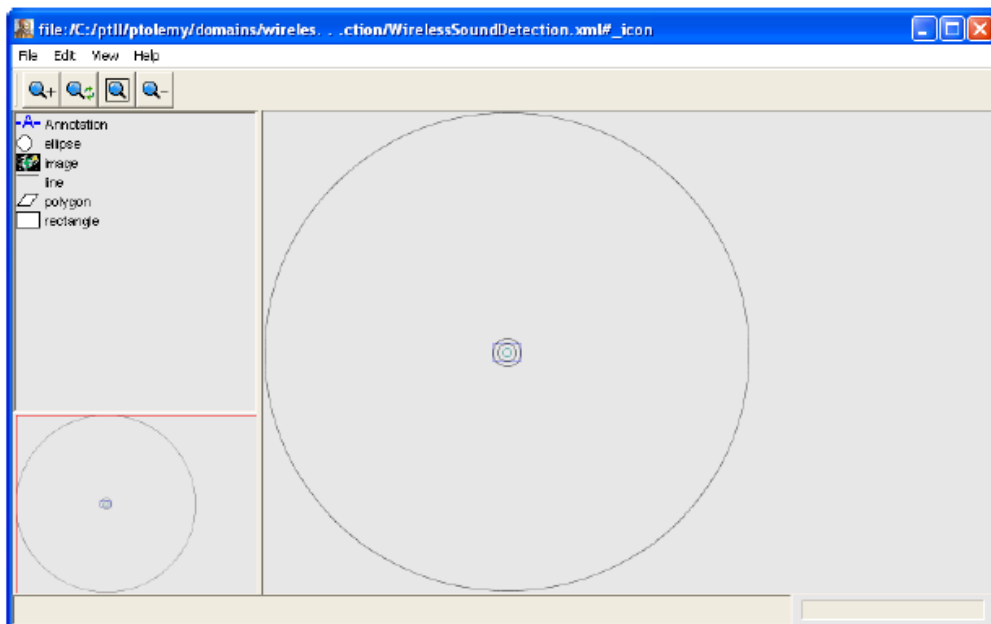
1) نمایش بصری (آیکون ها) :

ابتدا اکتور منبع صوت را بررسی کنید. در ابتدا توجه کنید که چه طور آیکون آن موقعی که پارامتر *soundRange* را تغییر می دهید ، عوض می شود. تعریف آیکون می تواند بوسیله راست کلیک روی آیکون و انتخاب "Edit Custom Icon" نمایش داده شود (ویرایش شود). توجه کنید که برای این اکتور ، شما باید ماوس را روی خط دور یکی از دایره های متمرکز قرار دهید . ره ظاهر شده در شکل 15 نمایش داده شده است. به یاد داشته باشید که فقط بخش مرکزی آیکون قابل Zoom Fit (همچنانکه در شکل 15) برای دریافت تصویر کامل

کلیک کنید (شکل 16 را ببینید). کتابخانه در سمت چپ می تواند برای اضافه کردن آیتم ها به آیکون استفاده شود. به دایره بیرونی توجه کنید، اندازه آن موقعی که پارامتر *soundRange* تغییر داده شود ، تغییر پیدا می کند. کلیک کنید (یا راست کلیک کنید و Configure را انتخاب کنید)، پنجره پارامتر در شکل 17 ظاهر می شود. توجه کنید که *height width* بوسیله عبارات با مقدار " $soundRange * 2$ " زبان اصطلاحی که می تواند اینجا استفاده شده است قدرتمند است.



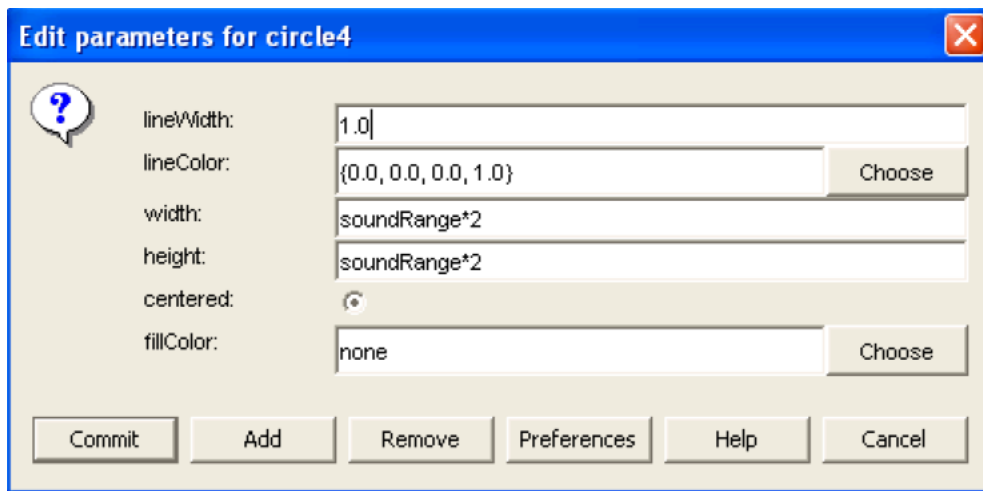
شکل (15) "edit custom icon" بعد از کلیک راست روی منبع صوت



شکل (16) نتیجه کلیک روی Zoom fit

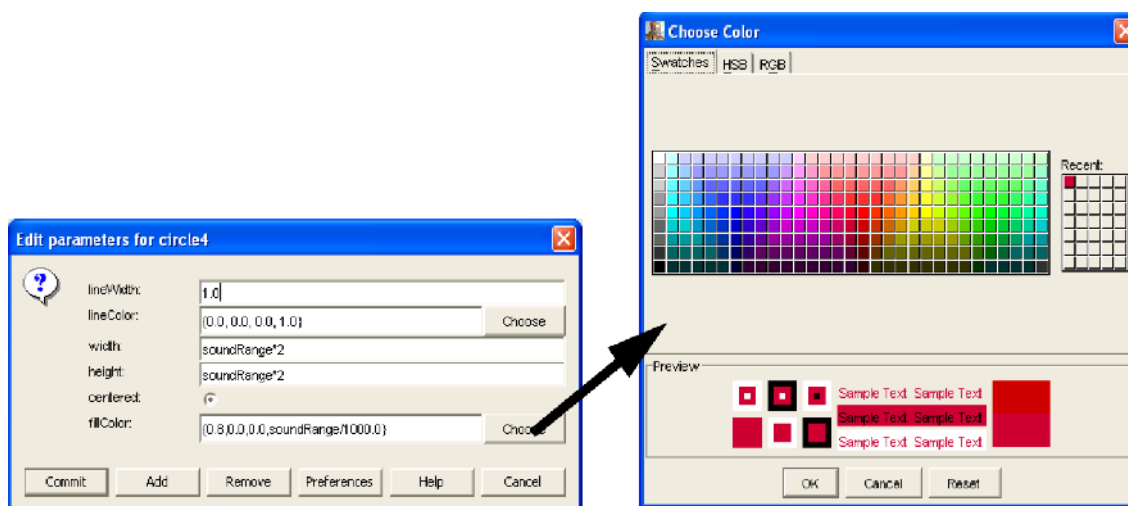
ما می توانیم دایره بیرونی با یک رنگ نیمه شفاف که درجه ماتی آن به پارامتر $soundRange$ بستگی دارد (شکل 18) در این شکل انتخاب کننده ()

$alpha$

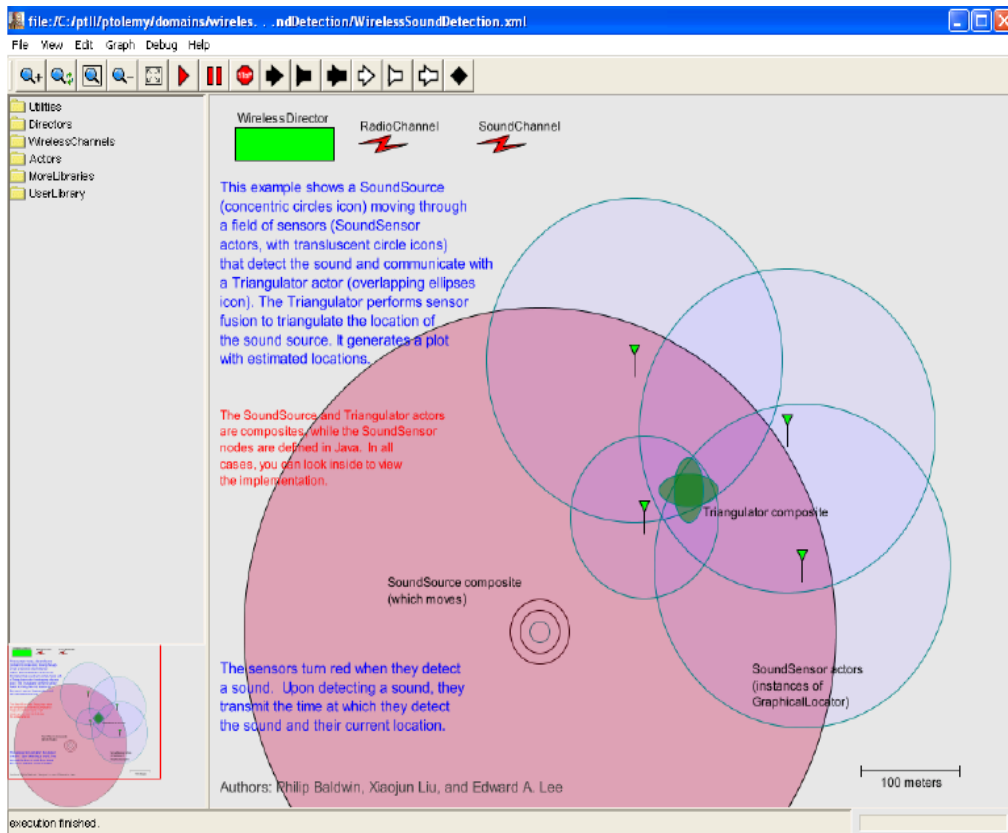


شکل (17) پارامترهای دایره بیرونی اکتور منبع صوت

که چهارمین عنصر از آرایه تعریف کننده رنگ است، به طور دستی با "soundRange/1000.0" تنظیم شد. نتیجه در شکل 19



شکل (18) تنظیم fill color دایره بیرونی منبع صوت که به SoundRange بستگی دارد



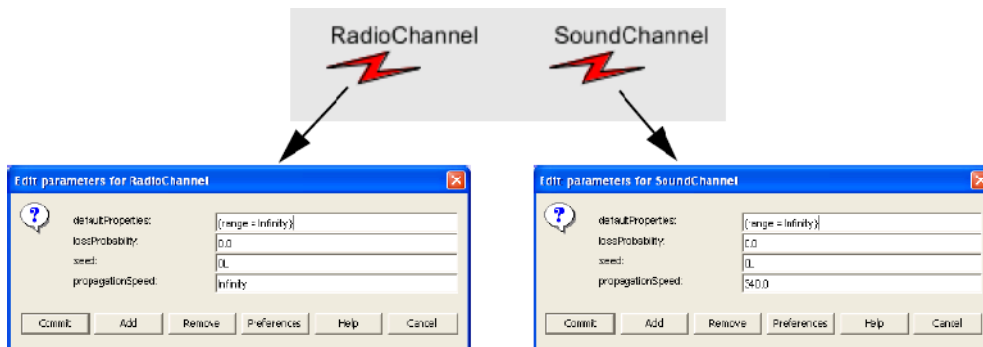
شکل (19) نتیجه تغییر رنگ دایره بیرونی منبع صوت

2) کانال ها :

مدل نشان داده شده در شکل 12 دو مدل کانال دارد که در شکل 20 با اضافه پارامترهایشان نشان داده شده است. شما می توانید ببینید که تنها اختلاف میان این دو کانال ($propagationSpeed$) برای کانال یویی ، "Infinity" تنظیم می شود، در حالی که برای کانال صوتی ، "340.0(meters/second)" تنظیم می شود.

توجه کنید که هر دو کانال یک پارامتر به نام $defaultProperties$ $\{range=Infinity\}$. این عبارت یک $record$ با یک فیلد به نام "range" "Infinity" تعیین می کند. فیلد های پارامتر $defaultProperties$ از یک کانال، مسیر هایی را که در یک ارسال ویژه می تواند به طور منحصر به فرد سفارشی شود ، تعیین کند. در این مورد ، یک ارسال ویژه از طریق هر کانال می تواند به طور اختیاری یک برد را تعیین کند. تعیین نشود ، آنگاه پیش فرض استفاده می شود، که بی نهایت⁴⁵ ، تعیین می کند که هیچ محدودیتی وجود ندارد. یک ارسال در رسیدن به گیرنده موفق خواهد بود و اهمیت ندارد که چقدر از گیرنده دور است.

⁴⁵ Infinity

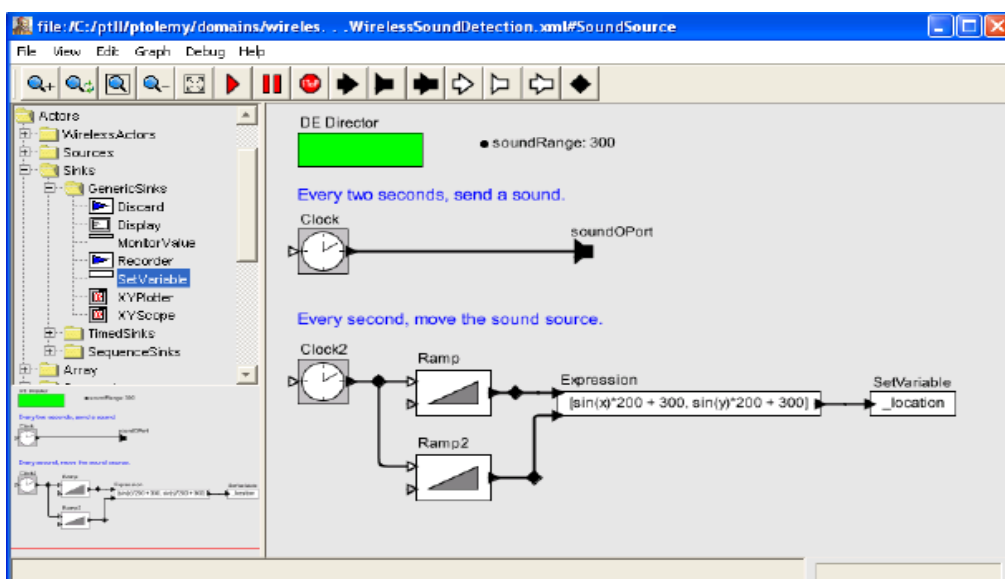


شکل (20) کانال شکل 12 و پارامتر هایش

3) اکتورهای مرکب :

ما دیدیم که چه طور می توانیم نمایش ویزوالی از یک اکتور را سفارشی کنیم. چه طور ما می توانیم رفتار آن را تعیین کنیم؟ اکتور منبع صوت در شکل 12 به طور واقعی یک *composite actor* است که رفتار آن به وسیله مدل Ptolemy مشخص می شود. برای تشخیص این تعریف، روی اکتور راست کلیک کنید و *Look Inside* را انتخاب کنید. مدل داخلی در شکل 21 نمایش داده شده است.

منبع صوت مرکب در شکل 21 یک DE Director دارد که این مدل را به عنوان یک مدل رویداد گسسته از Ptolemy II تعریف می کند. DE به خوبی با مدل های بی سیم کار می کنند. بنا بر این DE در گره های بی سیم عادی است. *DE Director soundRange* پیش فرض 300 نمایش بخش بالایی که یک رویداد صوتی را می فرستد، و یک بخش پایینی که آیکن را حرکت می دهد.



شکل (21) نتیجه Look Inside اکتور منبع صوت در شکل 12

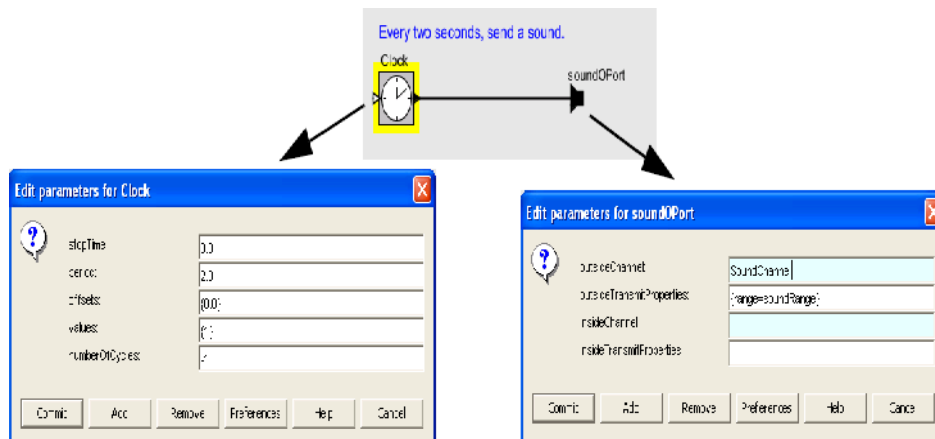
در ابتدا به بخش بالاتر توجه کنید. آن یک کلاک⁴⁶ و یک *port* که "soundPort" نامیده می شود دارد. (شکل 22) پارامترهای هم کلاک و هم پورت بوسیله دابل کلیک روی آنها (یا راست کلیک روی آنها و انتخاب *configure*) می آیند، همچنین در شکل نشا . به یاد داشته باشید که *period* کلاک با 2.0 تنظیم می شود و *values* {1} تنظیم می شود، یک آرایه تک عنصری با عدد صحیح 1. این اشاره دارد به این که کلاک باید یک صوت را هر دو ثانیه تولید کند. مقدار تولید شده عدد صحیح 1 است که هیچ معنی خاصی ندارد.

soundPort نیز پارامترهایی دارد، همچنانکه در شکل 22 *outsideChannel* . یک پارامتر با مقدار رشته ای است که مقدار "SoundChannel" این نام کانالی است که این پورت برای ارسال استفاده خواهد کرد، و باید مطابق با نام کانال نشان داده شده در شکل 20 .

outsideTransmitProperties "range=soundRange" را دارد که یک *record* با یک فیلد "range" ، که به سادگی مقدار از پارامتر *soundRange* ،

اکتور مرکب بدست می آید. باشید که این کار مقدار پیش فرض بی نهایت را برای این فیلد ، بنابراین پارامتر *soundRange* فقط ظاهر ویژوالی آیکون را کنترل نمی کند، بلکه برد ارسال را نیز کنترل می کند. به منظور تصمیم گیری که آیا گیرنده در برد هست یا خیر، همه نمونه های ایجاد شده ت *Visualsense* موقعیت آیکون را به عنوان یک نمایشی از موقعیت گره استفاده می کنند. بخش هایی اختیاری اند .

اگرچه همه این نمونه ها موقعیت دو بعدی را استفاده می کنند، ساختار نرم افزار بیان شده موقعیت سه بعدی را نیز پشتیبانی می کند.

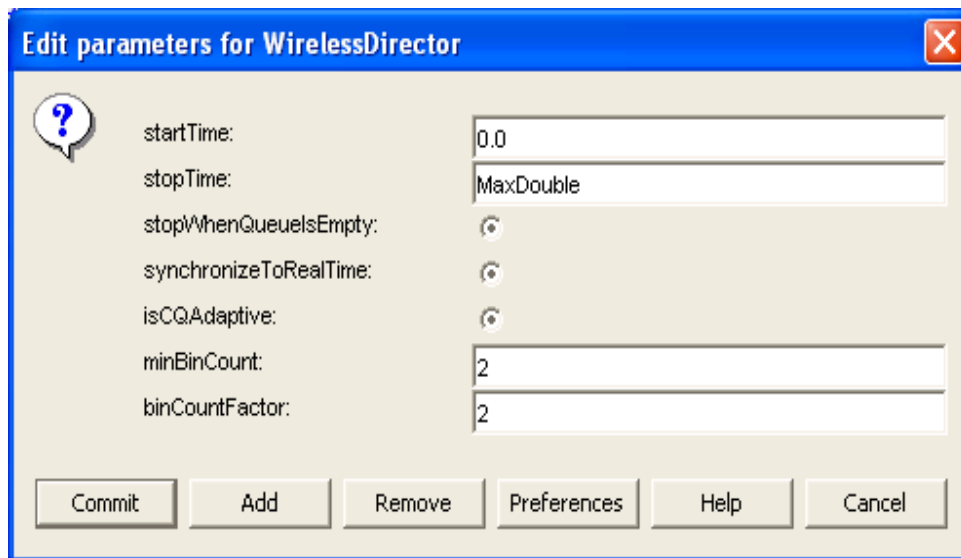


شکل (22) بخشی از مرکب در شکل قبلی که رویداد صوتی را تولید می کند

4) کنترل اجرا :

در شکل 12 مؤلفه ای است که اجرای مدل را کنترل می کند. مانند اغلب مدل ها آن نیز پارامترهایی دارد پارامترهای آن در شکل 23 . توجه کنید که زمان توقف⁴⁷ "MaxDouble"

⁴⁶ Clock
⁴⁷ stop time



شکل (23) wireless director در شکل 12

تنظیم می شود، که عدد بسیار بزرگ $1.7976931 \times 10^{308}$ این تعیین می کند که مدل باید برای همیشه اجرا گردد. توجه کنید که پارامتر *synchronizeToRealTime* Director کنترل می شود. این بدین معنی است که هنگامی که مدل را اجرا می کنید، اکتور کلاک که یک صوت را در هر 2 ثانیه یکبار تولید می کند، اجازه داده نخواهد شد تا رویداد هایی را در سرعت بیشتر ایجاد کند. این پارامتر برای دریافت مقیاس های زمانی واقع گرایانه هنگام اجرا استفاده می . این پارامتر باید برای مدل های در حال اجرا کنترل شود. دیگر پارامترهای Director درباره میزان سازی عملکرد شبیه ساز رویداد گسسته است. آنها فراتر از قلمرو این گزارش هستند.

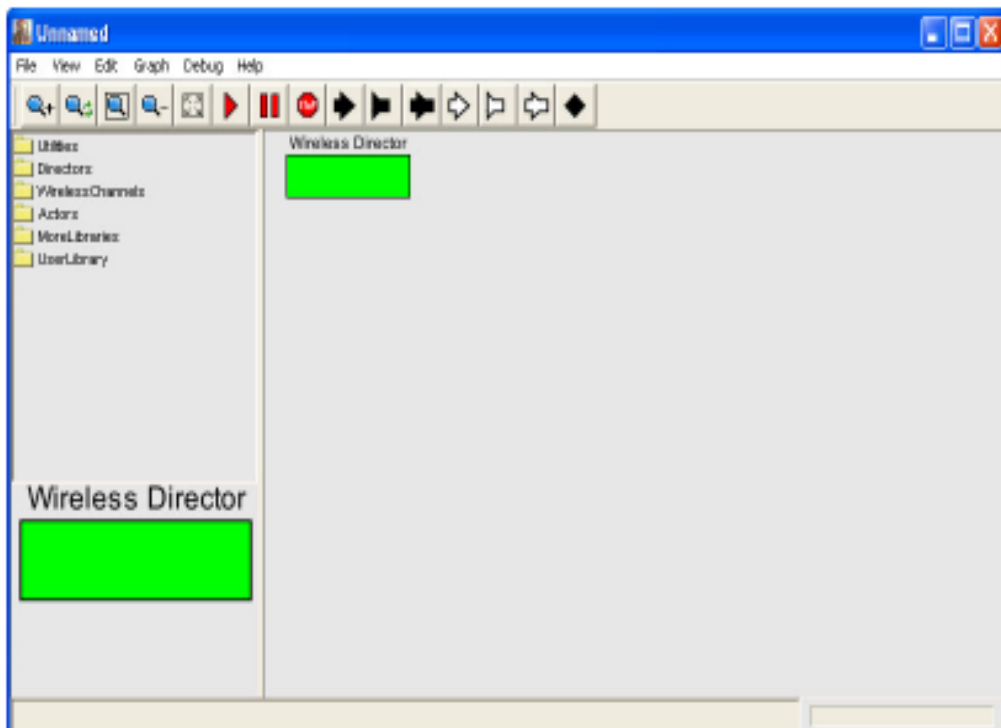
5) ساخت یک مدل جدید :

حالا ما می خواهیم به ایجاد یک مدل شبکه بی سیم حسگر جدید پردازیم. *Visualse* , *Graph Editor* را انتخاب کنید. این کار پنجره ای مانند آنچه در شکل 24 نشان داده شده را نتیجه می دهد. آن شامل یک *Wireless Director* ، و نه هیچ چیز دیگری . *PowerLossChannel* کتابخانه ی *WirelessChannel* در سمت چپ بکشید⁴⁸، همچنانکه در شکل 25 . به پارامتر های این کانال توجه کنید ، که در شکل 25 دقت کنید که پارامتر *defaultProperties* شامل یک رکورد به همراه دو فیلداست، *{range=Infinity, power=Infinity}*. این کانال می تواند برای مدل های مختلف در توان ارسال و همچنین اتلاف توان به عنوان تابعی از فاصله استفاده شود. مدلی را خواهیم ساخت که اگر گیرنده به اندازه کافی توان دریافت کند ارتباط را بدست می آورد، در غیر این صورت ارتباط را

Documentation برای اکتور *PowerLossChannel* (سایر اکتورها) می تواند بوسیله کلیک راست روی اکتور *Get Documentation* بدست بیاید. در این مثال ، ما صفحه نمایش داده شده در شکل 26 را خواهیم داشت که متن تولید شده برای کلاس جاوا که این کانال را مشخص می کند ، نشان می دهد. بالای این صفحه نمایش،

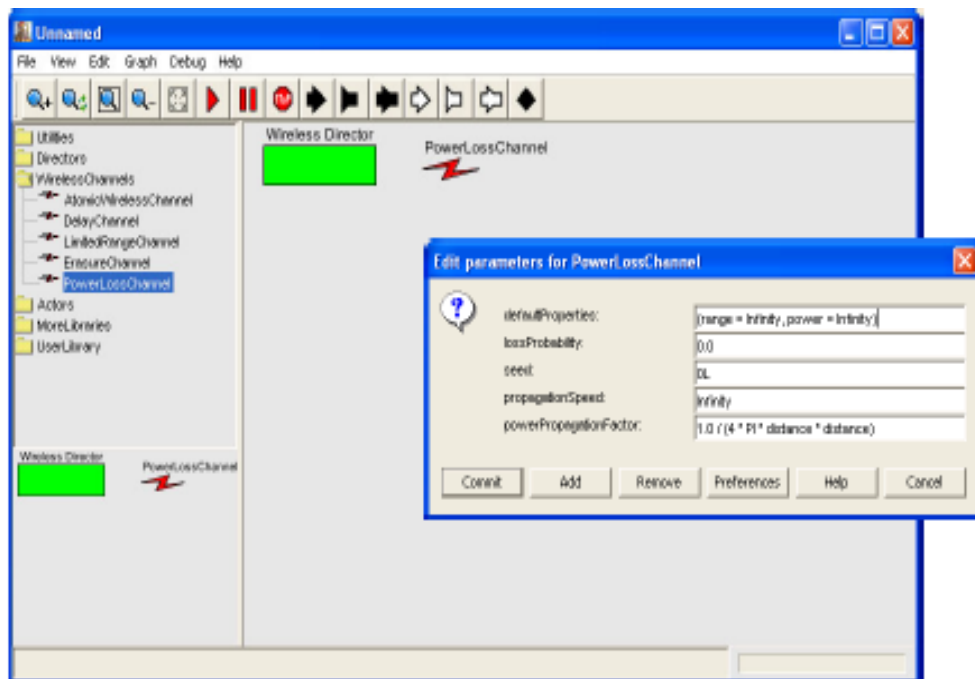
⁴⁸ Drag

زنجیره ای این اکتور نمایش داده شده که اشاره دارد به اینکه ، اکتور از `LimitedRangeCannel` توسعه یافته به همین ترتیب از `DelayChannel` و `ErasureChannel` و آن نیز از `AtomicWirelessChannel` گسترش می یابد. هر یک از این کانال ها مقدار کمی سودمندی را اضافه می کنند،⁴⁹ کد منبع⁴⁹ برای هر یک به عنوان یک مثالی از اینکه چه طور مدل های کانال را تعریف می کنند ، فراهم می شود. شما می توانید کد منبع را با راست کلیک و انتخاب `Look Inside` ببینید. (شکل 27). به عنوان مثال در شکل 27 ، `powerPropagationFactor` را توضیح می دهد.

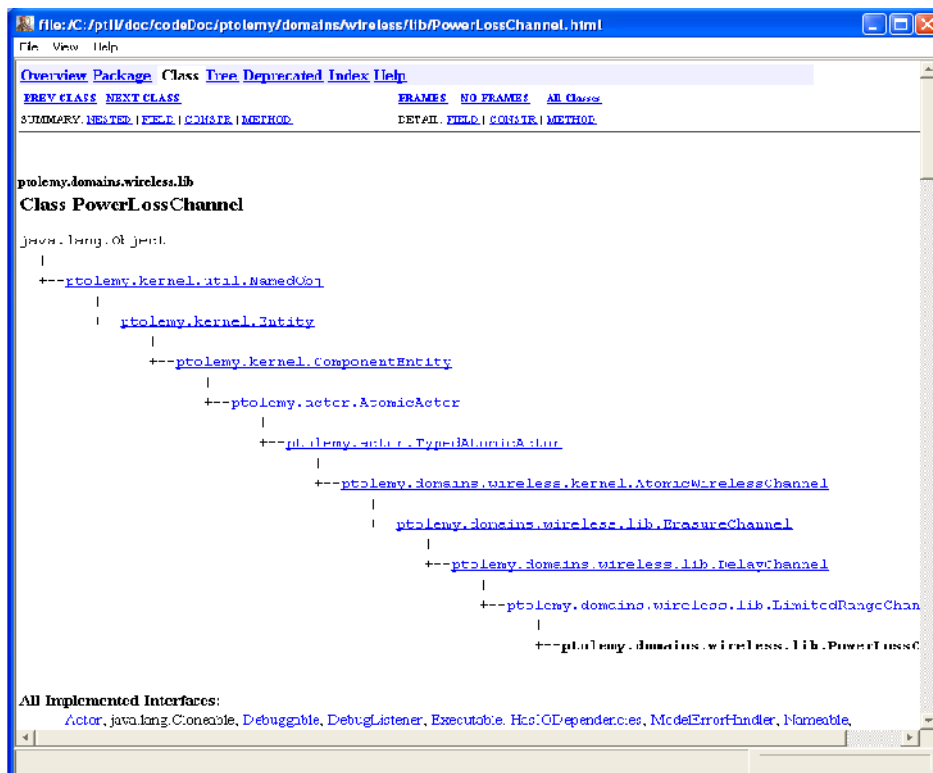


شکل (24) پنجره ساخت یک مدل جدید

⁴⁹ Source code



شکل (25) مدل جدید ثابت شده با یک کانال



شکل (26) PowerLossChannel documentation

"انتشار توان به عنوان یک عبارتی که ارزیابی می شود و سپس در فیلد توان از مشخصات ارسال قبل از رها سازی به گیرنده ضرب می شود. برای راحتی کار، یک متغیر که **distance** نامیده شده در دسترس است و معادل با فاصله میان فرستنده و گیرنده است موقعی که فرمول انتشار توان ارزیابی می شود. بنابراین، عبارت می تواند به این فاصله بستگی داشته

مقدار فیلد توان باید به عنوان توان در فرستنده تفسیر شود ولیکن به عنوان چگالی توان در گیرنده باید تفسیر یک گیرنده ممکن است چگالی توان را با بازدهی اش و یک قلمرو (افزایش دهد. یک گیرنده می تواند سپس توان بدست آمده را برای مقایسه در برابر یک آستانه قابل کشف، یا برای تصمیم گیری نسبت سیگنال به تداخل، کرد.

مقدار پیش فرض $powerPropagationFactor$ $1.0/(4*PI*distance*distance)$ فرض بر این است که توان ارسال به صورت یکنواخت در یک کره به شعاع $distance$ توزیع شده است. نتیجه افزایش این بوسیله یک توان ارسال چگالی توان (گیرنده باید این چگالی توان را بوسیله قلمرو حسگری که آن را برای گرفتن انرژی استفاده می کند، افزایش دهد (مثل ناحیه آنتن) و همچنین یک فاکتور بازدهی که نشان می دهد که چه طور آن به طور مؤثر انرژی را می گیرد.

```

file:///C:/ptll/ptolemy/domains/Wireless/lib/PowerLossChannel.java
File Help
/** A channel with a distance-dependent power loss.
Copyright (c) 2004 The Regents of the University of California.
All rights reserved.
Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
copyright notice and the following two paragraphs appear on all copies
of this software.
IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY
FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF
THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE
PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF
CALIFORNIA HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES,
ENHANCEMENTS, OR MODIFICATIONS.
PTOLEMY_VERSION_4
COPYRIGHTENDKEY
*/
package ptolemy.domains.wireless.lib;
import ptolemy.data.DoubleToken;
import ptolemy.data.RecordToken;
import ptolemy.data.ScalarToken;
import ptolemy.data.Token;
import ptolemy.data.expr.Parameter;
import ptolemy.data.type.BaseType;
import ptolemy.data.type.RecordType;
import ptolemy.data.type.Type;
import ptolemy.domains.wireless.kernel.WirelessIOPort;
import ptolemy.kernel.CompositeEntity;

```

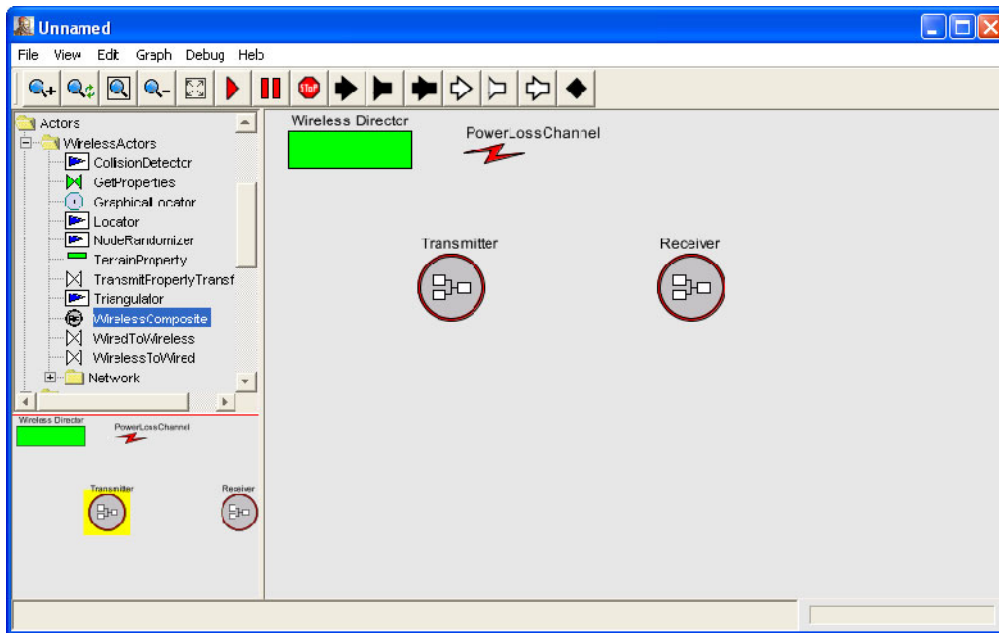
شکل (27) منبع کد برای PowerLossChannel

فیلد توان از مشخصات ارسال می تواند بوسیله فرستنده به عنوان رکورد با یک فیلد توان از نوع دابل⁵⁰ پشتیبانی شود. پیش فرض این کانال بی نهایت است."

به ما اجازه بدهید تا یک مدل را با استفاده از این پارامترها بسازیم.

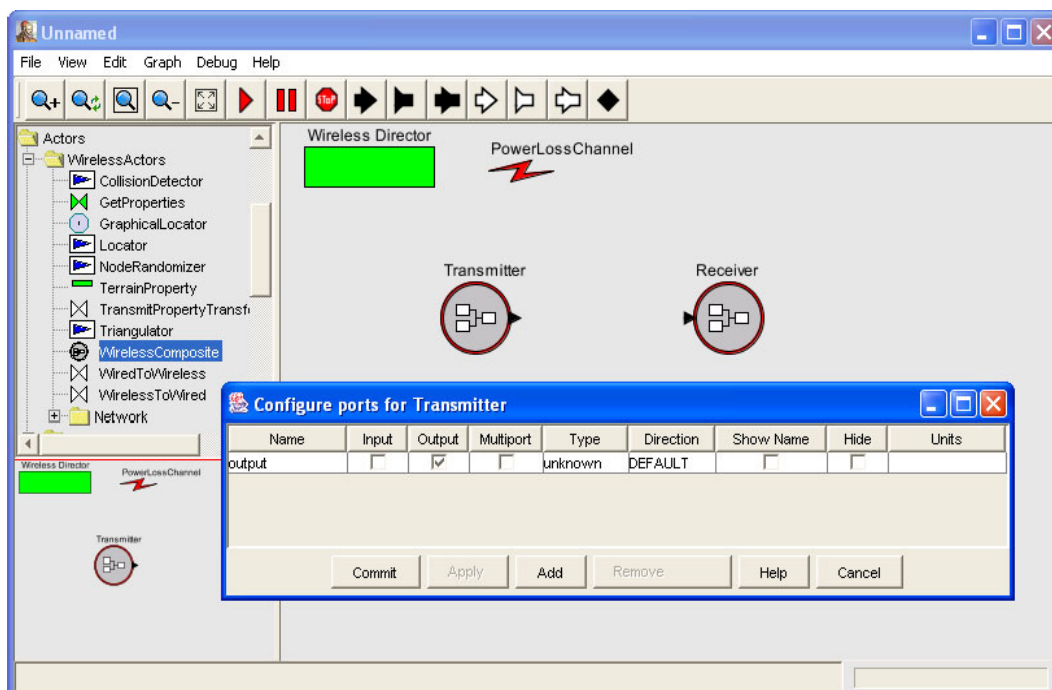
WirelessComposite را از کتابخانه Actor → WirelessActors چپ بکشید. Receiver Transmitter بوسیله کلیک راست روی آنها و انتخاب Customize Name تغییر نام دهید (شکل 28). این مؤلفه ها به پورت نیاز دارند. برای ایجاد آن روی آنها کلیک راست کنید و Configure Ports انتخاب نمایید. روی دکمه ADD کلیک کنید و یک پورت خروجی با نام output برای فرستنده ایجاد کنید و یک پورت

⁵⁰ Double



شکل (28) wirelesscomposite

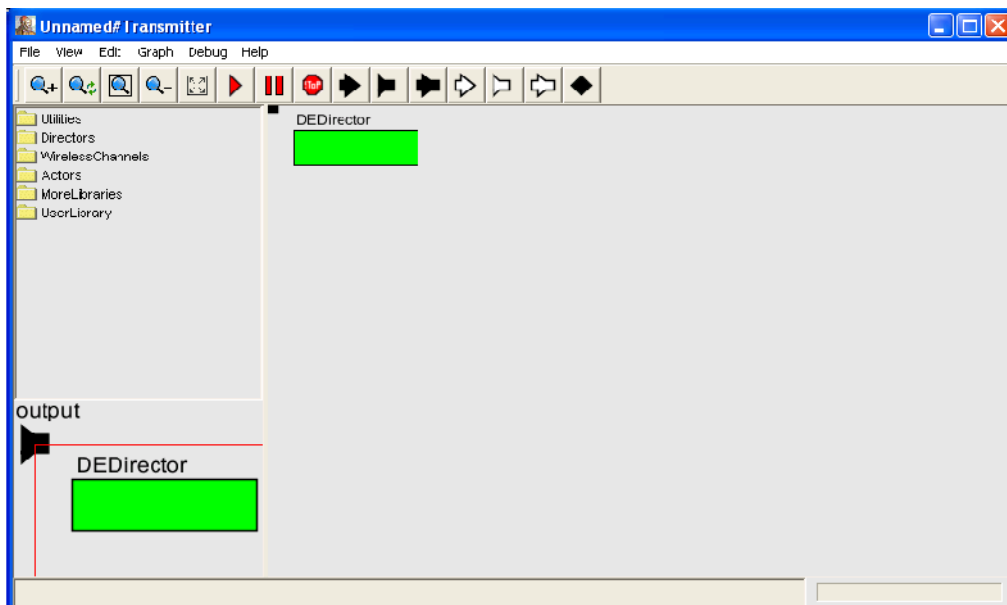
برای *input* گیرنده ایجاد کنید (شکل 29). روی هر پورت کلیک راست کرده و *configure* را انتخاب کنید *outsideChannel* "PowerLossChannel" مشخص کنید (این قسمت باید واقعا با نام کانال منطبق
(.



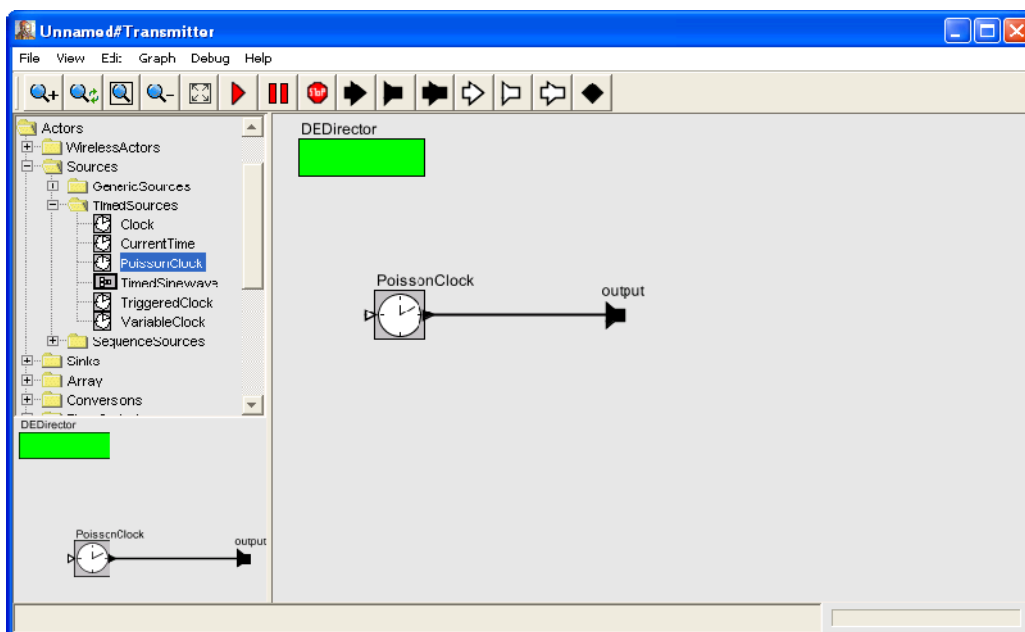
شکل (29) مدل با پورت های اضافه شده به فرستنده و گیرنده

ما کار را با ساکن کردن فرستنده و گیرنده با مدل های ساده از گره ها شروع می کنیم. برای انجام این کار، *Look Inside* می کنیم، که پنجره شکل 30 حاصل می شود. توجه کنید که پورت خروجی در بالا سمت چپ واقع شده

آن را به یک مکان قابل قبول تر انتقال دهید و به آن یک نمونه از اکتور PoissonClock از کتابخانه Actors → Source → TimedSource را اتصال دهید (شکل 31). برای ساخت یک اتصال، از پورت خروجی اکتور PoissonClock کلیک کنید و بکشید، یا Ctrl+Click کرده و از پورت خروجی فرستنده به پورت خروجی اکتور PoissonClock بکشید. اکتور PoissonClock رویدادهایی را در زمان های تصادفی تولید می کند، در جایی که زمان بین رویدادها از یک متغیر تصادفی نمایی با ابزار داده بوسیله پارامتر *meanTime* از پواسون کلاک بدست می آید. مقدار پیش فرض آن 1.0 است که برای اهداف ما مناسب است. اگر به پنجره بالاتر برگردید و روی WirelessDirector دابل کلیک کنید تا پارامتر *synchronizeToRealTime* آن را ست کنید، رویدادهایی را با نرخ متوسط یک در ثانیه تولید خواهد کرد.



شکل (30)



شکل (31) فرستنده تکمیل شده

اکتور گیرنده را Look Inside کنید و مدل نشان داده شده در شکل 32 را بسازید. اکتور Ramp می تواند در کتابخانه اکتورها تحت مسیر زیر پیدا کنید:

Source→SequenceSource

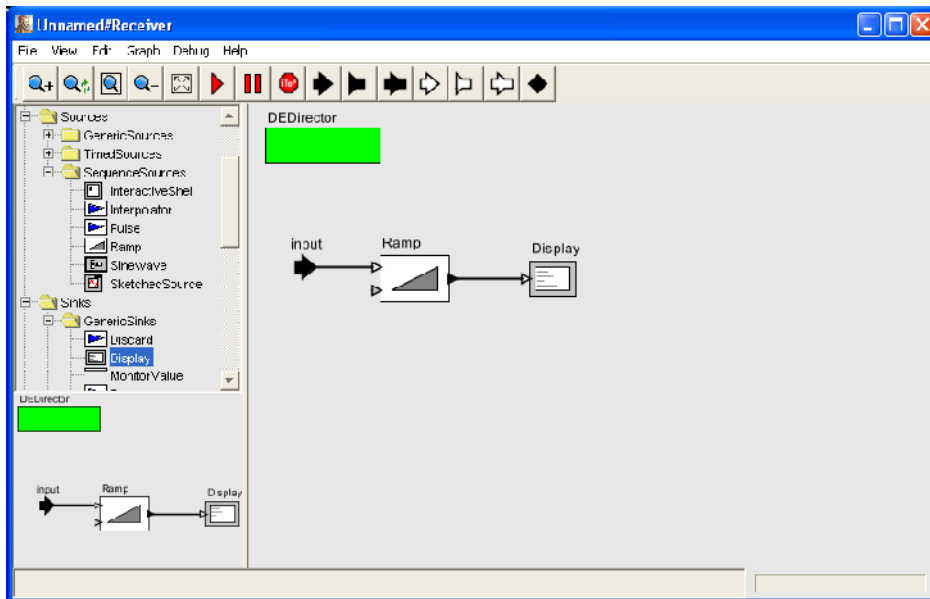
و اکتور display مسیر زیر بیاید:

Sinks→GenericSinks

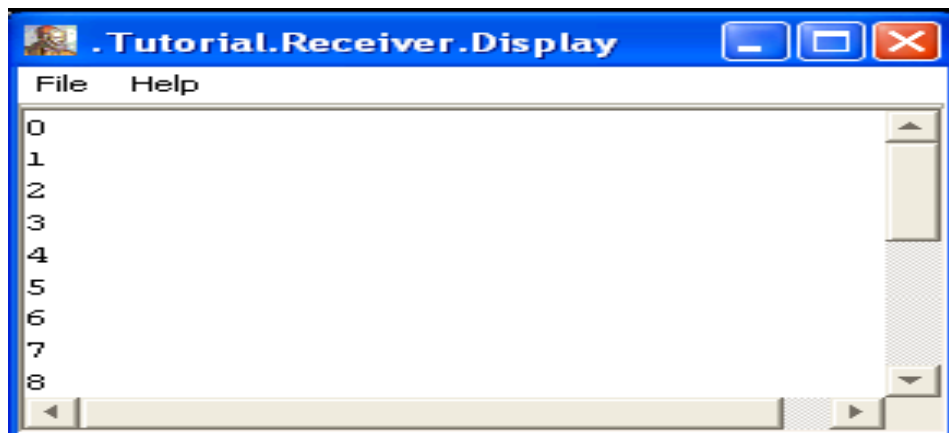
همچنانکه در سمت چپ شکل نشان داده شده است. حال مدل آماده اجرا می باشد. روی مثلث قرمز رنگ در نوار ابزار کلیک کنید نتیجه اجرا در display در شکل 33

به یاد داشته باشید که پارامتر *synchronizeToRealTime* را ست کنید، سپس تعدادی عدد در زمانهای تصادفی با فاصله میانگین یک ثانیه ظاهر خواهد شد.

شما ممکن است بخواهید تا فایل خود را ذخیره کنید (File→Save). xml (یا moml) استفاده کنید تا مطمئن شوید که Visualsense این را به عنوان یک فایل مدل تشخیص خواهد داد.

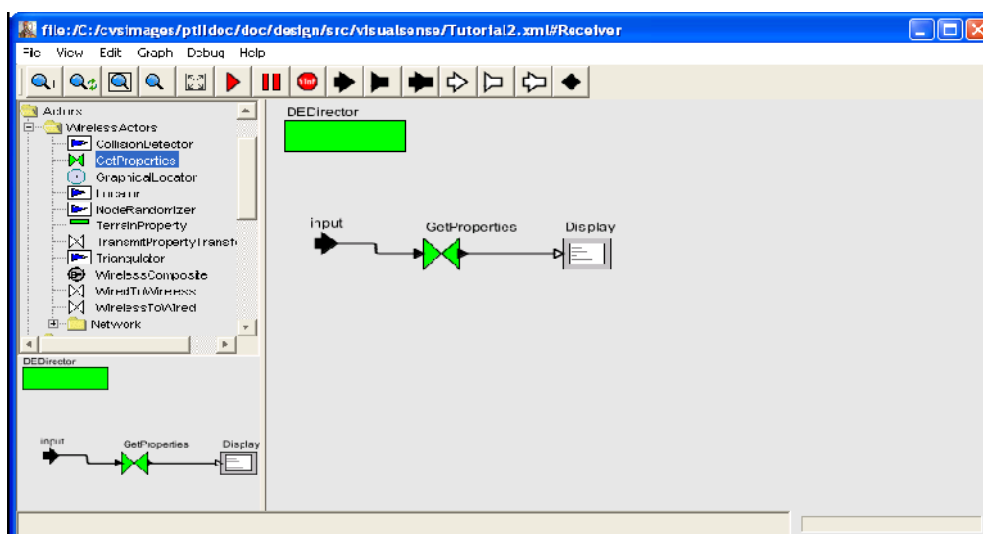


شکل (32) گیرنده تکمیل شده

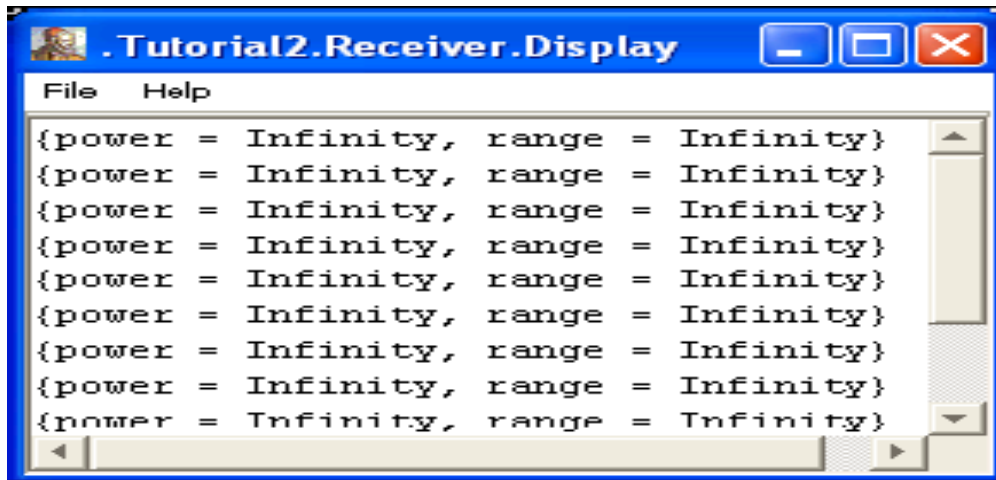


شکل(33) display که نتیجه اجرای توضیح داده شده بالا را نمایش می دهد

اجازه دهید که این مدل را برای اینکه اتلاف توان کانال به عنوان تابعی از فاصله مشاهده شود، اصلاح نماییم. این کار، اکتور `GetProperties` در کتابخانه `WirelessActor` → `Actor` پیدا کنید، و جایگزین اکتور `Ramp` داخل گیرنده کنید (شکل 34). حال مدل را اجرا کنید نتایج در `display` در شکل 35 توجه کنید که توان دریافت شده همیشه بی نهایت است، که بسیار مفید نیست.

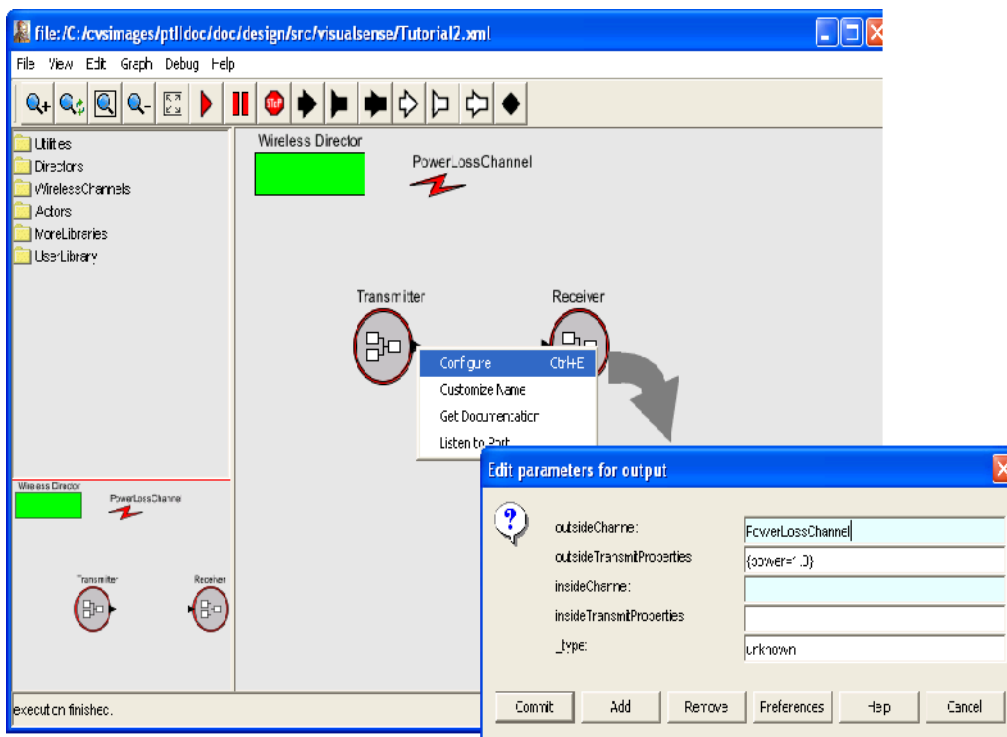


شکل(34) گیرنده اصلاح شده که مشخصات دریافت شده را نمایش می دهد.



شکل (35) display که نتیجه استفاده از گیرنده طراحی شده در بالاست.

فرستنده توان ارسال را مشخص نکرده است، PowerLossChannel یک توان پیش فرض بی نهایت همانند شکل 25. معرفی شده توسط کانال نامربوط است، زیرا در این مدل توان ارسال بی نهایت است و موقعی که اتلاف غیر صفر افزوده می شود، باز هم توان بی نهایت حاصل می شود. برای آنکه به یک مدل معقول تری از اتلاف انرژی دست پیدا کنیم توان انتقال را بوسیله کلیک راست روی پورت خروجی تنده و تنظیم پارامتر *outsideTransmitProperties* "{power=1.0}" تنظیم کنید (شکل 36) مجدد شبیه شکل 37، که تغییر پذیری سطح توان بوسیله حرکت گیرنده در اطراف فرستنده در حالیکه مدل در بدست می آید.



شکل (36) تنظیم توان ارسال فرستنده

```

File Help
{power = 3.1084949822636E-6, range = 1
{power = 4.0549030087107E-6, range = 1
{power = 4.360409399778E-6, range = Ir
{power = 4.7017708446646E-6, range = 1
{power = 5.5262133018019E-6, range = 1
{power = Infinity, range = Infinity}
{power = 0.0031830988618, range = Infi
{power = 3.1830988618379E-4, range = 1
{power = 1.2242687930146E-4, range = 1

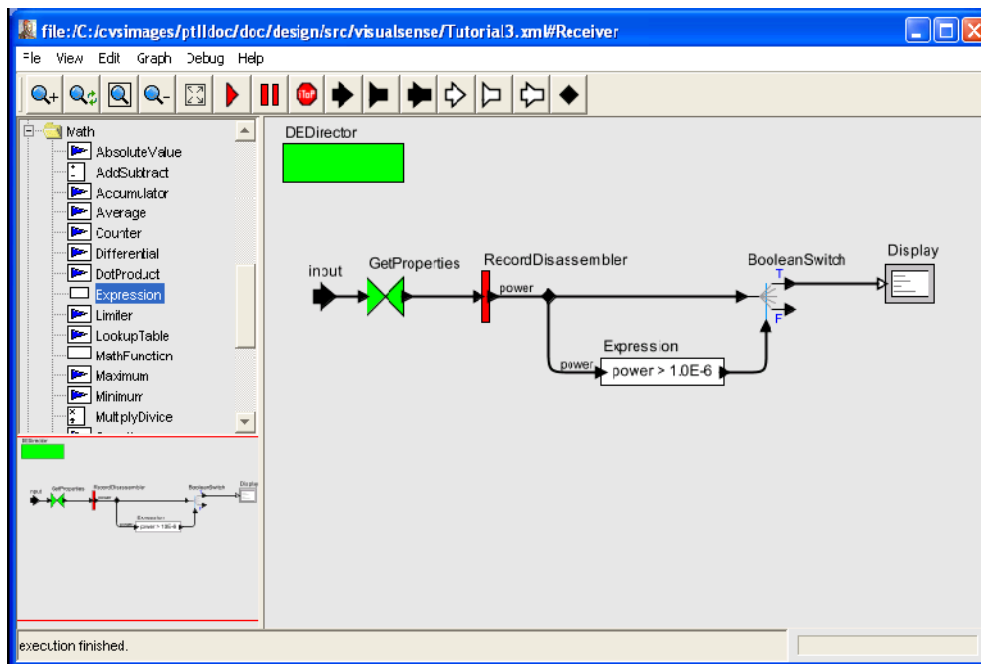
```

شکل (37) `display` که نتیجه استفاده از مجموعه توان ارسال را در شکل بالا نشان می دهد.

به شکل 37 توجه کنید که یکی از مقادیر توان دریافت شده بی نهایت دارد. این موقعی رخ می دهد فرستنده و گیرنده بر یکدیگر منطبق باشند. `PowerLossChannel Documentation` انی کنید که مقدار فیلد `Power` در مشخصات دریافت شده یک چگالی توان است، نه یک توان مطلق. از این رو اگر فرستنده و گیرنده یک فضای فیزیکی را اشغال کنند و فرستنده یک منبع نقطه ای باشد، آنگاه چگالی توان در گیرنده بی نهایت است. یک مدل گیرنده این چگالی توان را بوسیله یک سطح آنتن دهی مؤثر و یک بازده آنتن دهی افزایش خواهد داد تا یک سطح توان دریافت شده مطلق را دریافت کند.

چگالی توان دریافت شده می تواند برای تصمیم گیری در گیرنده که آیا ارسال موفق بوده یا خیر استفاده گردد. این کار مدل گیرنده را برای دریافت ساختار نمایش داده در شکل 38 اصلاح می کنیم. اکتور های استفاده شده در مسیرهای زیر قرار دارند:

RecordDisassembler:Actors→FlowControl→Aggregators
 Expression:Actors→Math
 BooleanSwitch:Actors→FlowControl→BooleanFlowControl



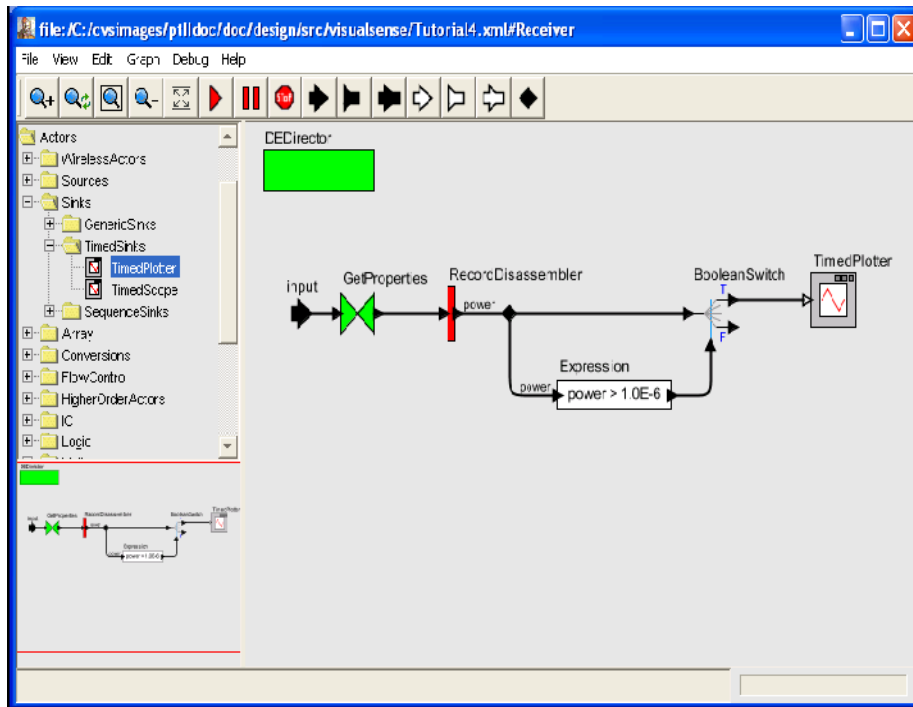
شکل(38)مدل گیرنده که رویداد را صرف نظر می کند در جایی که توان زیر مقدار آستانه باشد

اکتور RecordDisassembler فیلد هایی از یک رکورد را استخراج می کند. شما باید پورت های خروجی را که همان نام فیلد را دارند، که در این مورد Power است را ایجاد کنید. برای استفاده از اکسپریشن Expression باید پورت های ورودی را ایجاد کنید، و هر نامی را که می خواهید استفاده کنید ("Power" در شکل 38)، و سپس یک عبارت را که خروجی ها را در قالب ورودی ها مشخص می کند، بدهید (در شکل 38 "Power > 1.0E-6"). خروجی اکتور Expression (true) ، دریافت شده بزرگتر از مقدار بیان شده در Expression ، غیر این صورت اشتباه (false) . این سیگنال Boolean control Booleanswitch می کند، که ورودی اش را به یکی از دو خروجی بسته به مقدار ورودی کنترل می فرستد. در این مورد ما خروجی true ، که مقادیر توان دریافت شده بیش از مقدار بیان شده در Expression است را مشاهده می کنیم. توجه کنید که در شکل 38 بعضی اتصالات شامل یک لوزی سیاه کوچک هستند، این مکانیسم ویژگی برای مسیر دهی یک سیگنال به چندین مکان است. برای ایجاد یک چنین لوزی (که vertex نامیده می شود) با کلیک ctrl+click یا استفاده از نوار ابزار آن را ایجاد کنید و برای ایجاد یک اتصال از آن کلید ctrl ، روی لوزی کلیک کرده و اتصال را به مکان دلخواه بکشید و آن را رها کنید.

6) به کارگیری اکتور plot :

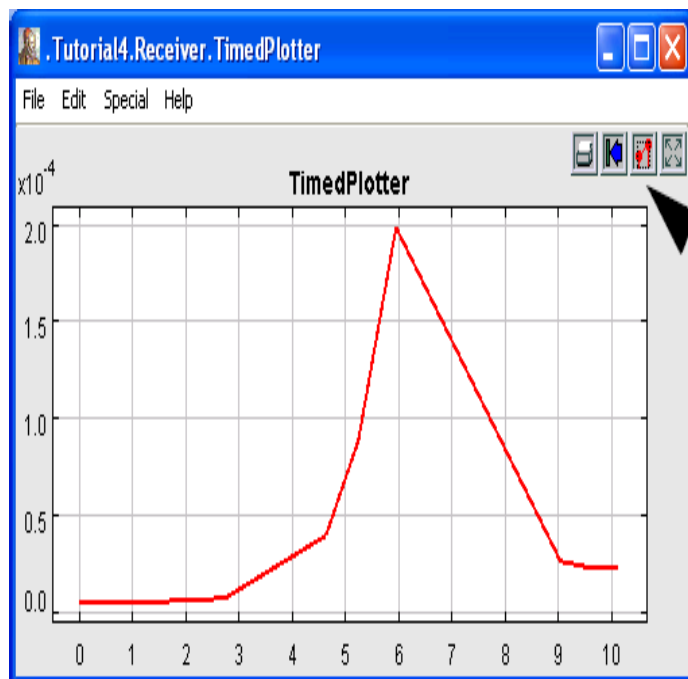
اغلب برای یک مدل در ترسیم داده ها نسبت به نمایش آن در حالت متنی مفید است. مدل شکل 38 شده در شکل 39 اصلاح کنید، که در آنجا اکتور Display بوسیله Timedplotter از مسیر زیر جایگزین شده است.

Actors → Sink → TimedSinks



شکل (39) گیرنده توان دریافت شده را به صورت تابعی از زمان رسم می کند.

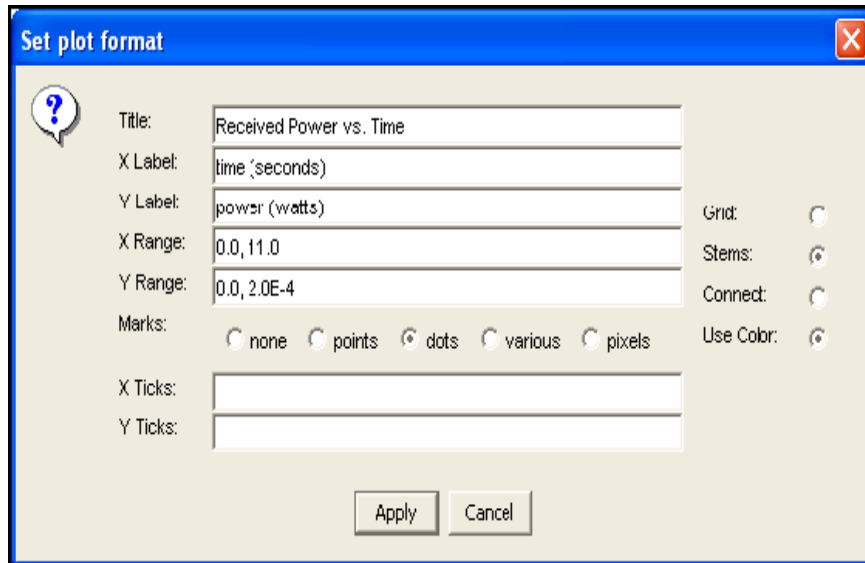
نتیجه اجرا در شکل 40 نمایش داده شده است.



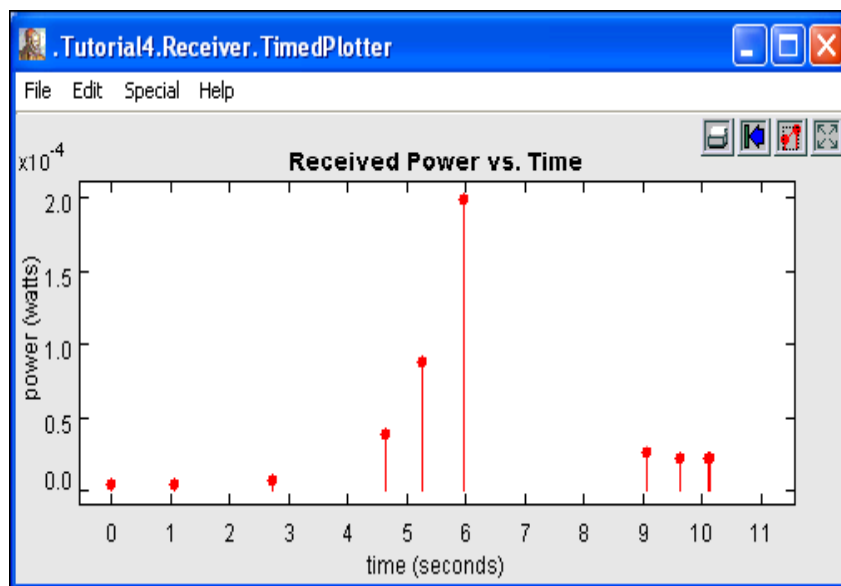
format button

شکل (40) نمودار نشان دهنده توان دریافت شده به صورت تابعی از زمان

صفحه نمایش نمودار به طور قابل توجهی می تواند بهبود پیدا کند. **format** روی دکمه **plot** راست کلیک کنید. (شک 40) تا پنجره شکل 41 پارامترها را همچنانکه در پنجره نتایج نشان داده شده در **plot** در شکل 42 تنظیم کنید. **zoom** کردن می توانید از دکمه های بالای آن استفاده کرد و یا از کلیک کردن روی **display** و کشیدن آن استفاده کرد.



شکل (41) پنجره تنظیم فرمت ن



شکل (42) نمودار تغییر کرده با استفاده از پنجره بالا

قابلیت های مدل سازی :

Visualsense یک گسترشی از مدل ساز رویداد گسسته از Ptolemy II. آن به طور گسترده ای معنای رویداد گسسته را حفظ می کند، اما مکانیسم اتصال مؤلفه ها را تغییر می دهد، بطوریکه سیم بندی آشکار نیاز نیست.

شده در بخش قبلی، مدل های سیمی و بی سیم به طور سلسله مراتبی ترکیب می شوند، Ptolemy II که یک مجموعه بسیار قدرتمندی از مکانیسم های مدل سازی است، می تواند برای ساخت مدل های گره های حسگر و اثرات

در این بخش، ما معانی رویداد گسسته را به طور صریح شرح می دهیم، و مدل کانالی را که برای تصمیم گیری اتصالات در شبکه های حسگر و مدل مؤلفه سلسله مراتبی را برای هر گره حسگر بحث می کنیم. ما سپس قابلیت ها را با مطرح کردن بعضی از مثال ها که بهبود یافته اند به عنوان نمونه هایی با سیستم، توضیح می دهیم.

1) شبیه سازی رویداد گسسته :

Director یک نقش کلیدی در Ptolemy II بازی می کند: *semantics* را از یک مرکب⁵¹ تعریف می کند. مدل همزمانی (concurrency) و مکانیسم ارسال را ارائه می دهد. Director, Visualsense شبیه ساز را پیا پیا می کند. Wireless Director یک زیر کلاس تقریباً تغییر نکرده از پیش موجود DEDirector Ptolemy II

2) مدل های کانال :

یک مدل کانال در Visualsense خودش یک اکتور است. موقعی که یک فرستنده رویدادی را در یک پورت بی سیم که کانال را بوسیله نام ارجاع می ، تولید می کند رویداد در کانال برای ارسال رها می شود. کانال ممکن است مشخصات را که بوسیله فرستنده پشتیبانی می شود را تغییر دهد، و ممکن است تاخیر رویدادی را به یک فرستنده در مدل تاخیر انتشار رها سازی کند. Visualsense، ضمانت کانال در آنجا پایان می یابد. دیگر مؤلفه ها برای مدل کردن اثرات عوارضی و محیط از جمله بهره آنتن دهی و غیره استفاده شود.

3) مدل های گره بی سیم :

گره های حسگر در جاوا می توانند مدل شوند یا به طور جالب تری از مدل های DE (به عنوان بلوک دیاگرام) یا دیگر مدل های Ptolemy II (از قبیل مدل های DataFlow، ماشین های حالت متناهی یا مدل Continuous-time) استفاده کنند. Modal می تواند بوسیله طرح یک ماشین حالت متناهی و تهیه پالایش ها برای هر یک از حالت ها تعریف شود تا رفتار گره را در آن حالت مشخص کند. این برای مثال می تواند در مدل سازی مصرف انرژی عنوان تابعی از حالت استفاده شود و مدل های ساختگی از تزویج میان مصرف انرژی و Mac Protocol را ممکن می

4) مثال هایی از قابلیت مدل سازی :

اکثر قابلیت های مدل سازی اینجا با مثال شرح داده می شوند، که از پنجره ابتدایی در دسترس است.

4 1

Ptolemy II شامل یک سیستم نوع پیچیده است که شامل انواع پیوسته مانند رکورد ها ست. در بالا ما نشان دادیم که چه طور رکورد ها می توانند برای مشخصات ارسال استفاده شوند. آنها همچنین می توانند برای ساخت بسته ها با بار مفید

⁵¹ Composite

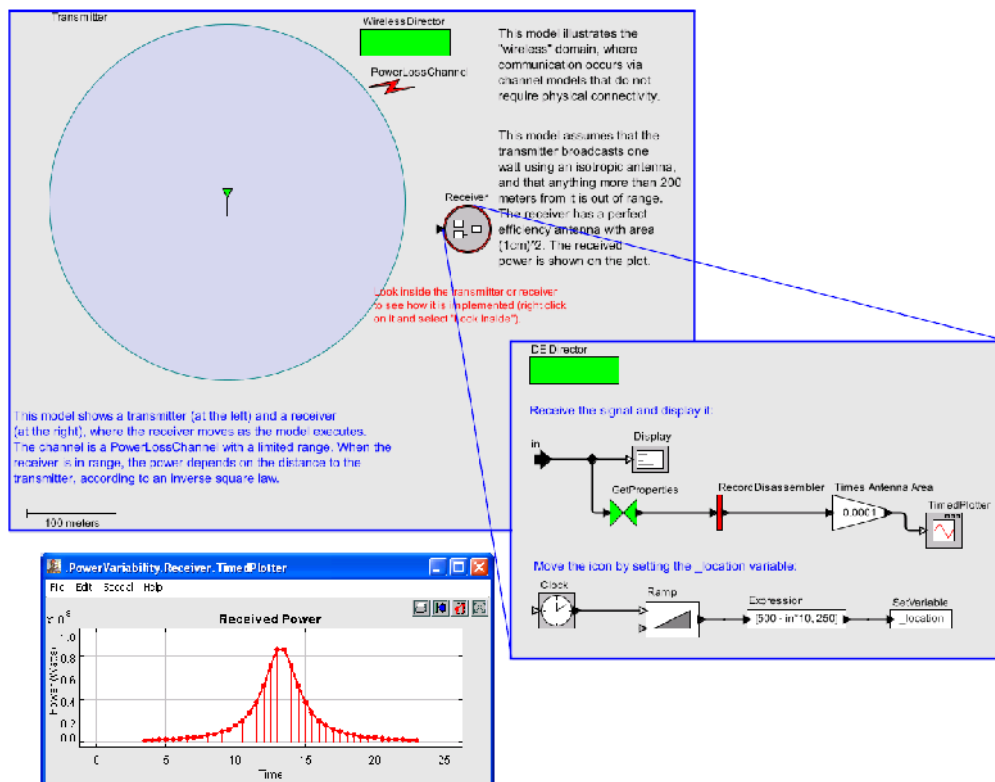
RecordAssembler , RecordDisassembler , مکانیسم یکسان است. ,
RecordUpdater می تواند برای مونتاژ و مجزا کردن رکورد ها استفاده شود.

: (4 2

ErasuerChannel , که یک کلاس پایه برای اغلب مدل های کانال است ,
LossProbability را که می تواند برای مدل کردن مستقل , لاف بسته های توزیع شده به طور مساوی استفاده شود, پیشنهاد می کند.

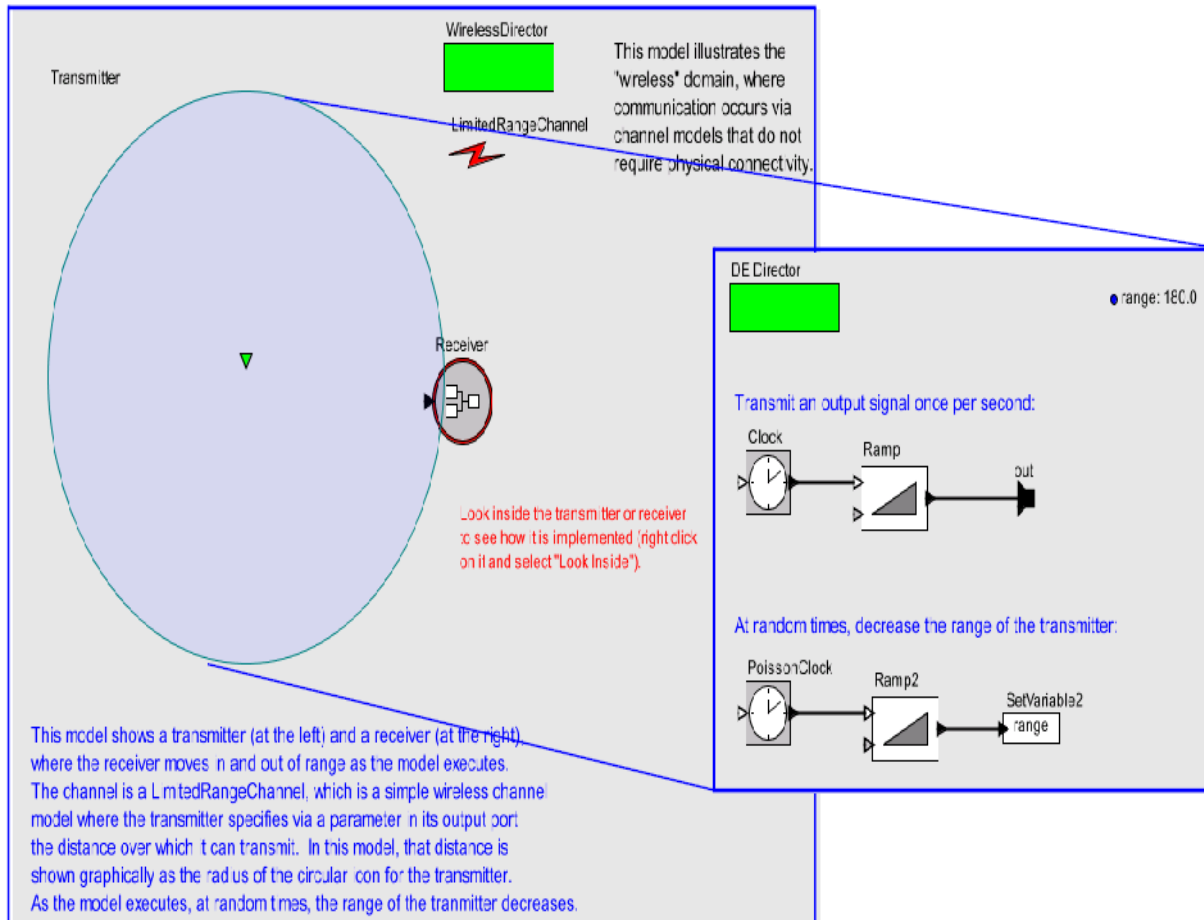
: (4 3

از آنجایی که گره های حسگر در شبکه های بی سیم بوسیله مدل های اختیاری تعریف می شوند, ایجاد مدل های مصرف انرژی یا توان آسان است. یک مثال ساده تحت عنوان "Circular Range Channel" در شکل 43 ,
که در سمت راست شما می توانید فرستنده ای را که یک پوآسون کلاک را برای کاهش برد ارسال استفاده می کند تا کاهش برد ارسال در طول زمان را همچنانکه باتری آن تخلیه می شود, مدل کند.



شکل (43)

اولیه شامل یک مدل به نام "Power Loss Channel" که تغییر توان در گیرنده را به عنوان تابعی از فاصله توضیح می دهد. پیاده سازی گیرنده و یک نمودار منتج از اجرای آن در شکل 44 این مدل همان قاعده را به عنوان مثال خود آموز توضیح داده شده در بالا استفاده می کند.



شکل (44)

در معانی رویداد گسسته اساسی *Visualsense*، رویدادها بلا فاصله در یک زمان بخصوص رخ می دهند. بدین معنی که آنها یک دوره زمانی (*duration*) برای مدل کردن برخورد پیام ها که زمان گیرند و کانال عمومی را تسهیم⁵³ می کنند صریحاً طول پیام⁵⁴ را لحاظ کند.

یک مثال ساده از چنین مدلی در شکل 45. در این مدل دو فرستنده یک کانال را تقسیم می کنند و پیام های با طول ثابتی را در زمان های تصادفی ارسال می کنند. چنانچه مدل را اجرا کنید یکی از فرستنده ها در الگوی چرخشی حرکت می کند. دور از فرستنده دیگر شروع به حرکت کرده و نزدیک آن می شود و از آن دور می شود. در هنگام شروع موقعی که از گیرنده دور است پیام هایش در صورتی توسط گیرنده دریافت می شود که فرستنده دیگر پیامی را که باعث همپوشانی در یک زمان نشود را ارسال نکند. چه پیام از فرستنده دیگر دریافت گردد چه برخوردی روی دهد بستگی به این دارد که فرستنده اولی چقدر دور است. اگر آن به اندازه کافی دور است پس توان تداخل⁵⁵ برای جلوگیری از ارسال کافی نیست بنابراین پیام دریافت می شود. اگر آن نزدیک تر باشد پس توان تداخل به اندازه کافی برای اینکه پیام دریافت نشود.

دو نمودار در شکل 45. نمودار بالایی پیام هایی را نشان می دهد که ارسال شده اند (در رنگ های آبی و)، یک نمایش ویژوالی از وقوع همپوشانی⁵⁶ را نشان می دهد. مقدار در نمودار توان دریافت شده را نشان می دهد، که ثابت است توان دریافت شده ثابت است. برای فرستنده متحرک توان دریافت شده شروع به کاهش می کند، سپس شروع به افزایش می کند تا تقریباً معادل با توان فرستنده ثابت می شود بعد از آن دوباره سقوط می کند. نمودار پایینی نشان می دهد که آیا پیام از بین رفته است یا خیر؟ در شکل در مجموع هفت پیام از بین رفته است. همه به جز یکی از پیام ()

طول پیام در این مدل با یک فیلد اضافی که به مشخصات ارسال⁵⁷ بوسیله کانال اضافه می شود نشان داده شده. پارامترهای این کانال در پایین سمت راست شکل 45. توجه کنید که پارامتر *defaultproperties* "{range=Infinity,power=Infinity,duration=1.0}" فیلد *duration* در این رکورد، طول پیام را نشان می دهد. فرستنده ها به طور اختصاصی می توانند این فیلد را نادیده گرفته و بوسیله تنظیم پارامتر *outsideTransmitProperties* از پورتهای شان یک *duration* دلخواه ایجاد کنند.

پیاده سازی گیرنده در شکل 46. در این مقدار سیگنال دریافت شده *Boolean*، که اگر تولید کننده پیام فرستنده ثابت باشد ارزش *false* خواهد داشت در صورتیکه صادر کننده *true*

اکتور *GetPropertyies* برای استخراج مشخصات دریافت شده استفاده می گردد، که شامل توان دریافت شده و طول پیام فیلد های *power* *duration* از رکورد مشخصات بوسیله *RecordDisassembler* استخراج می شوند و به *CollisionDetector* تحویل می گردند، تا تصمیم گیری کنند که چه پیام هایی دریافت شده و چه پیام هایی از بین بقیه مدل به ساخت نمودارها اختصاص می یابد.

اکتور *CollisionDetector* نسبتاً پیچیده است. متن توضیحی آن در شکل 47 نمایش داده شده است.

⁵² collisions

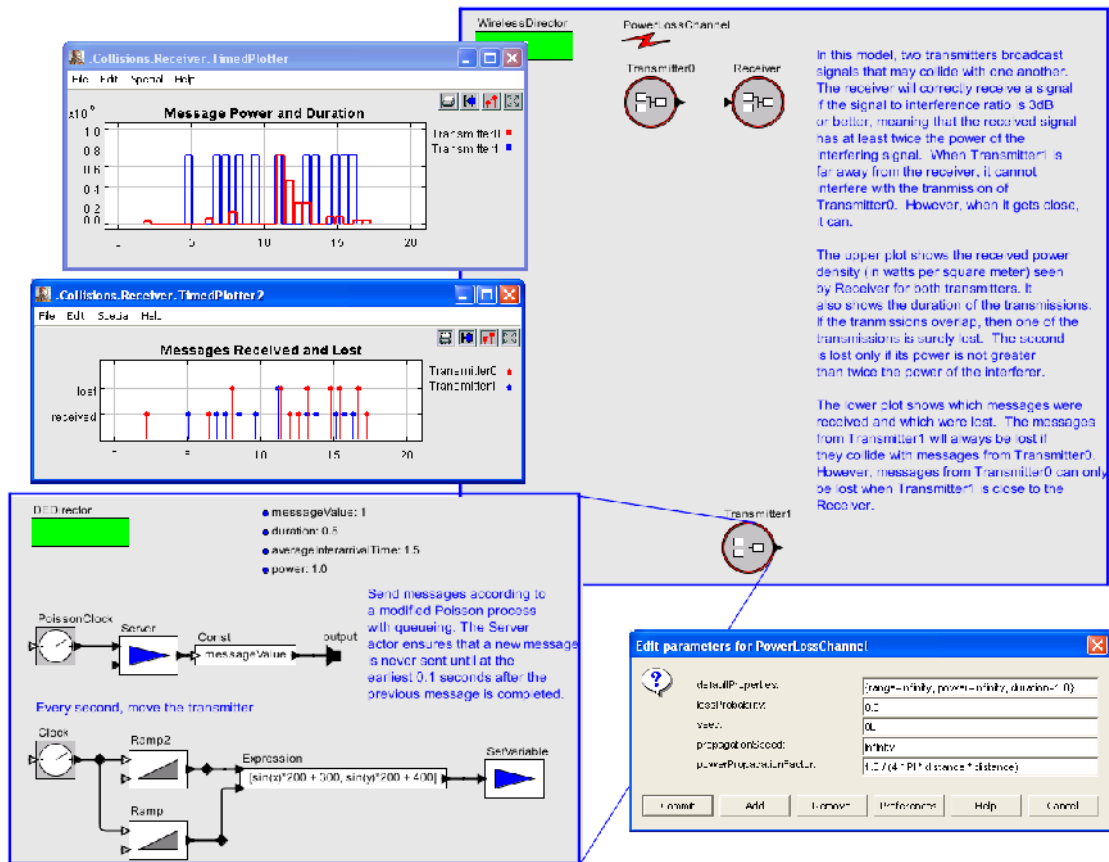
⁵³ Share

⁵⁴ message duration

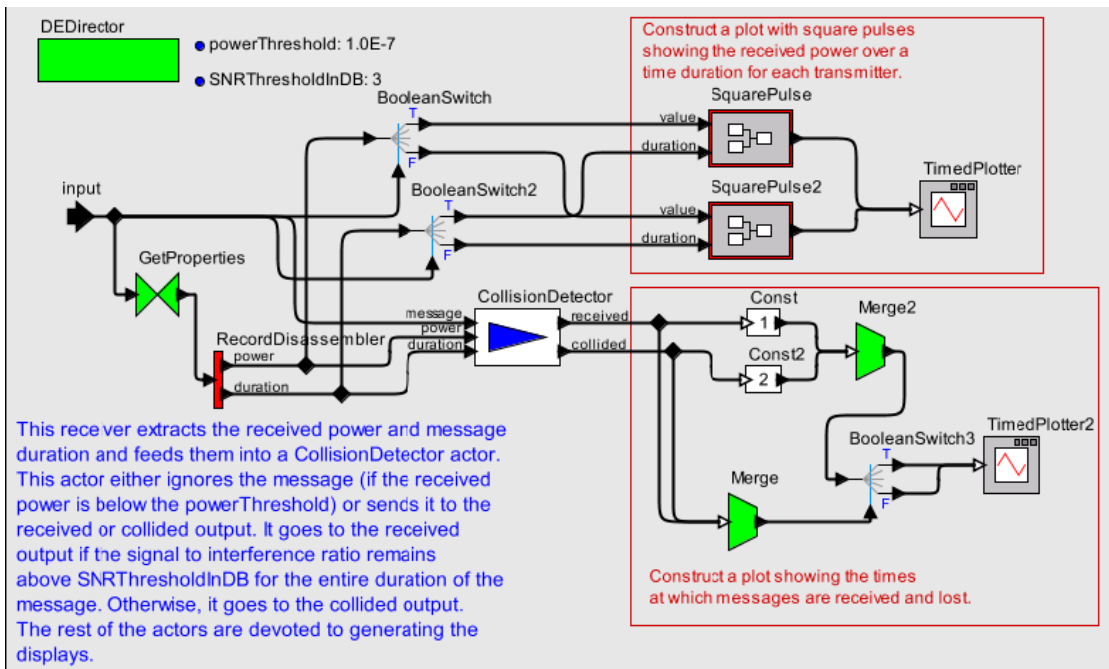
⁵⁵ interfering power

⁵⁶ Overlap

⁵⁷ transmit properties



شکل (45) مدل برخورد پیام ها که زمان گیر هستند



شکل (46) پیاده سازی گیرنده در شکل قبل

CollisionDetector: This actor models a typical physical layer front end of a wireless receiver. It models a receiver where messages have a non-zero duration and messages can collide with one another, causing a failure to receive. A message is provided to this actor at the time corresponding to the start of its transmission. Along with the message (an arbitrary token), the inputs must provide the duration of the message and its power. The message spans an interval of time starting when it is provided to this actor and ending at that time plus the duration. If another message overlaps with a given message and has sufficient power, then the given message will be sent to the *collided* output. Otherwise it is sent to the *received* output. In both cases, the message appears at the corresponding output at the time it is received plus the duration (i.e. the time at which the message has been completed).

The inputs are:

- *message*: The message carried by each transmission.
- *power*: The power of the received signal at the location of this receiver.
- *duration*: The time duration of the transmission.

The *power* and *duration* are typically delivered by the channel in the "properties" field of the transmission. The power is usually given as a power density (per unit area) so that a receiver can multiply it by its antenna area to determine the received power. It is in a linear scale (vs. DB), typically with units such as watts per square meter. The duration is a non-negative double, and the message is an arbitrary token.

The outputs are:

- *received*: The message received. This port produces an output only if the received power is sufficient and there are no collisions. The output is produced at a time equal to the time this actor receives the message plus the value received on the *duration* input.
- *collided*: The message discarded. This port produces an output only if the received message collides with another message of sufficient power. The output is produced at a time equal to the time this actor receives the message plus the value received on the *duration* input. The value of the output is the message that cannot be received.

This actor is typically used with a channel that delivers a properties record token that contains *power* and *duration* fields. These fields can be extracted by using a *GetProperties* actor followed by a *RecordDisassembler*. The *PowerLossChannel*, for example, can be used. However, in order for the type constraints to be satisfied, the *PowerLossChannel*'s *defaultProperties* parameter must be augmented with a default value for the *duration*. Each transmitter can override that default with its own message duration and transmit power.

Any message whose power (as specified at the *power* input) is less than the value of the *powerThreshold* parameter is ignored. It will not cause collisions and is not produced at the *collided* output. The *powerThreshold* parameter thus specifies the power level at which the receiver simply fails to detect the signal. It is given in a linear scale (vs. DB) with the same units as the *power* input. The default value is zero, i.e. by default it won't ignore any received signal.

Any message whose power exceeds *powerThreshold* has the potential of being successfully received, of failing to be received due to a collision, and of causing a collision. A message is successfully received if throughout its duration, its power exceeds the sum of all other message powers by at least *SNRThresholdInDB* (which as the name suggests, is given in decibels, rather than in a linear scale, as is customary for power ratios). Formally, let the message power for the *i*-th message be $p_i(t)$ at time *t*. Before the message is received and after its duration expires, this power is zero. The *i*-th message is successfully received if

$$p_i(t) > P \sum_{i \neq j} p_j(t) \quad (1)$$

for all *t* where $p_i(t) > 0$, where $P = 10^{(SNRThresholdInDB/10)}$, which is the signal to interference ratio in a linear scale.

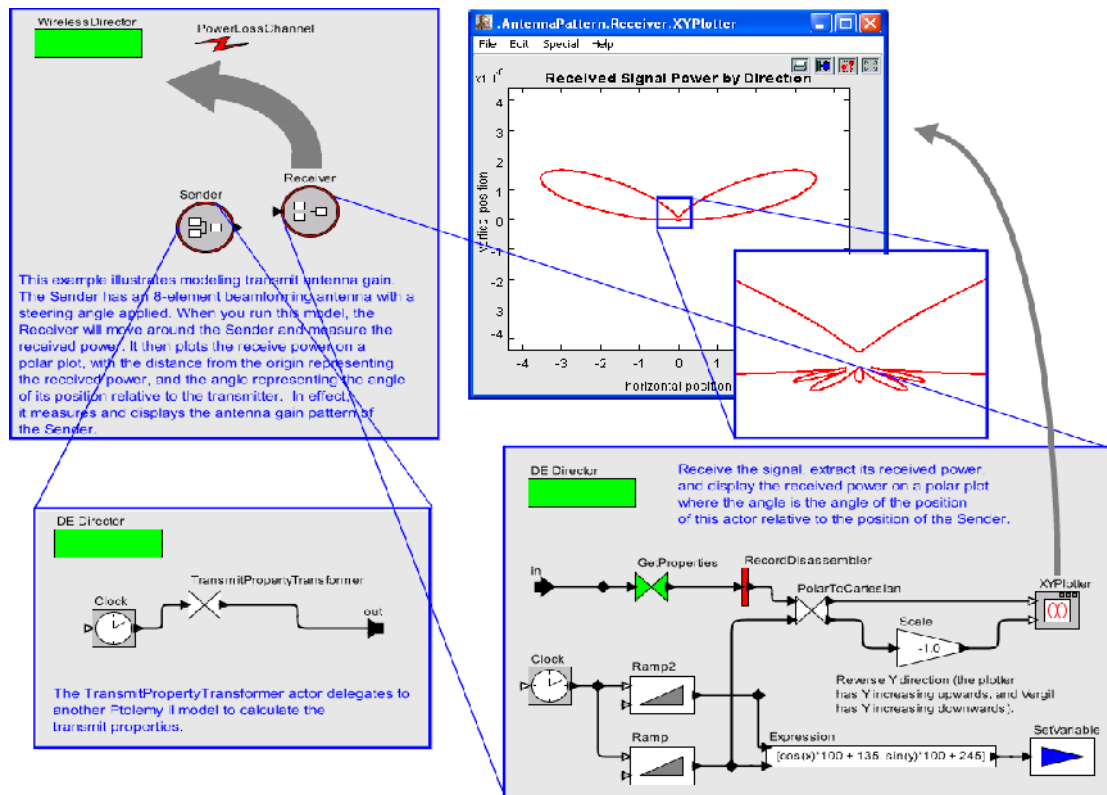
شکل(47) documentation برای اکتور collisiondetector

6 4 بهره آنتن دهی ارسال :

یک فرستنده برای یک کانال بی سیم یک آنتن جهت دار (directional antenna) . این یک پیچیدگی معنی داری را در مدل کننده مطرح می کند. اگر چه هدایت کنندگی یک خاصیت محلی از فرستنده است، اثر آن به موقعیت گیرنده بستگی . ما در بالا خاصیت ارسال را برای مدل سازی ائتلاف انتشار (propagation losses) دیده ایم.

ارسال همچنین برای بهره آنتن دهی استفاده می شود، فرستنده با کانال یک *property transformer* را ثبت می کند که اکتوری است که مشخصات ارسال را برای هر ارسال بخصوص اصلاح خواهد کرد. قبل از اینکه کانال رویدادی را به گیرنده تحویل دهد، *property transformer* را اجرا می کند و به آن اجازه می دهد تا مشخصات ارسال را اصلاح کند.

یک مثال ساده از یک مدل که شامل یک آنتن ارسال جهتی است در شکل 48 . این مدل در پنجره ابتدایی نرم افزار تحت عنوان "Transmit Antenna Gain" موقعی که این مدل را اجرا می کنید گیرنده در یک الگوی چرخشی اطراف فرستنده حرکت می کند و توان ارسال شده را اندازه گیری کرده و آن را ترسیم می کند. فرستنده یک آنتن با ترتیب مرحله به مرحله ای هشت عنصری با هدایت کننده⁵⁸ .



شکل (48) مدل شامل یک آنتن ارسال جهتی.

طراحی فرستنده کاملاً پیچیده است (شکل 49). آن با مثال نشان می دهد که چه طور توان مدل سازی کامل Ptolemy II می تواند در Visualese در بالا سمت چپ شکل، اکتور `transmitPropertyTransformer`

فرستنده را مدل می کند. رفتارش در ارسال بسیار ساده است: موقعی که با یک ورودی نشانه (token) ارائه می شود، این ارسال، این اکتور همان ورودی نشانه را بدون تغییر، در پورت خروجی تولید می کند. خودش را با کانال استفاده شده بوسیله پورتی که خروجی اش وصل به یک `Property Transformer` ثبت می کند. هنگامی که ارسال بی سیم از طریق پورت خروجی به بعضی گیرنده ها رخ می دهد، کانال بار دیگر `transmitPropertyTransformer` را برای هر گیرنده فراخوانی می کند، موقعیت گیرنده را فراهم می کند و مدل در بردارنده اکتور `transmitPropertyTransformer` را اجرا می کند.

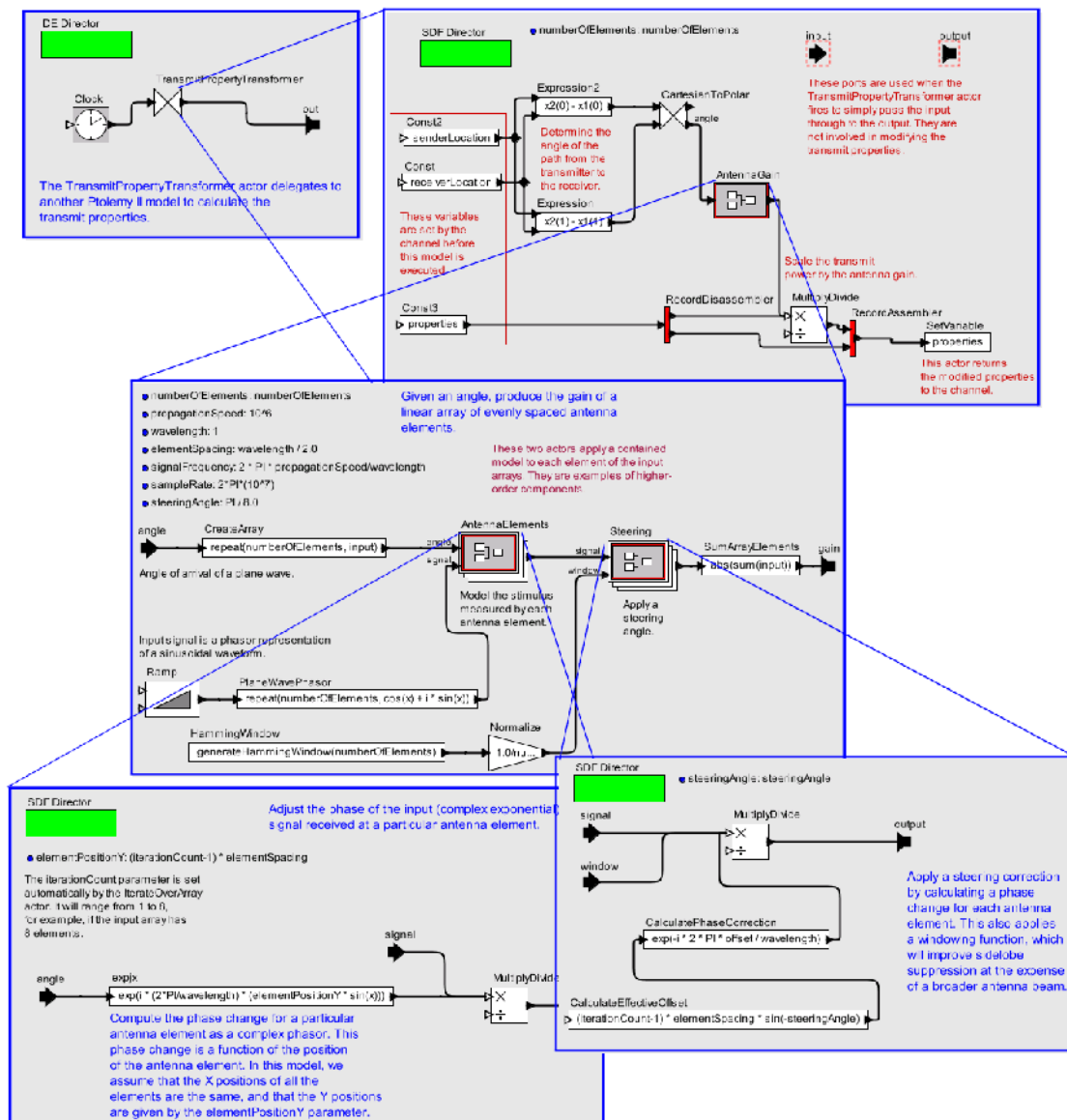
مدل در بردارنده اکتور `transmitPropertyTransformer` در شکل 49

شکل سطح بالای مدل نمایش داده شده است. آن نشان می دهد که موقعی که اجرا می شود (یکبار برای هر ارسال در پاسخ به کانال)، `properties receiverlocation`، `senderlocation` آماده می شود.

`properties` یک رکوردی است که در این مورد شامل یک فیلد `power` که بوسیله مدل برای محاسبه بهره آنتن دهی در هدایت از فرستنده به گیرنده اصلاح شده است. این مدل زاویه ارسال را محاسبه می کند،

جهت محاسبه می کند و سپس فیلد `power` از رکورد `properties` را مقیاس بندی می کند. توجه کنید این مدل یک `SDFDirector` `WirelessDirector` معمول که بیشتر به طور عمومی در `Visualese` استفاده می شد را نیز دارد. این بخاطر است که محاسبه بهره آنتن دهی اساساً یک تابع پردازش سیگنال

بهره آنتن دهی با استفاده از مدل نشان داده شده در میان شکل 49 محاسبه می شود. این مدل دو اکتور **IterativeOverArray** ("Steering" "ArrayElement") را برای مدل سازی عناصر آرایه و کاربرد بردار فرمان یا هدایت کننده استفاده می کند. این اکتور ها، اکتور هایی هستند که مدل های درون خود را یکبار برای هر عنصر آرایه ورودی اجرا می کنند. این اکتور ها مثال هایی از مؤلفه های *higher-order* ، ر این مورد قادر به تعریف یک مدل در جایی که تعداد عناصر آنتن بوسیله یک پارامتر بیشتر نسبت به سخت افزار داخل دیاگرام داده می همان مکانیسم برای مدل سازی بهره آنتن در گیرنده استفاده شود.



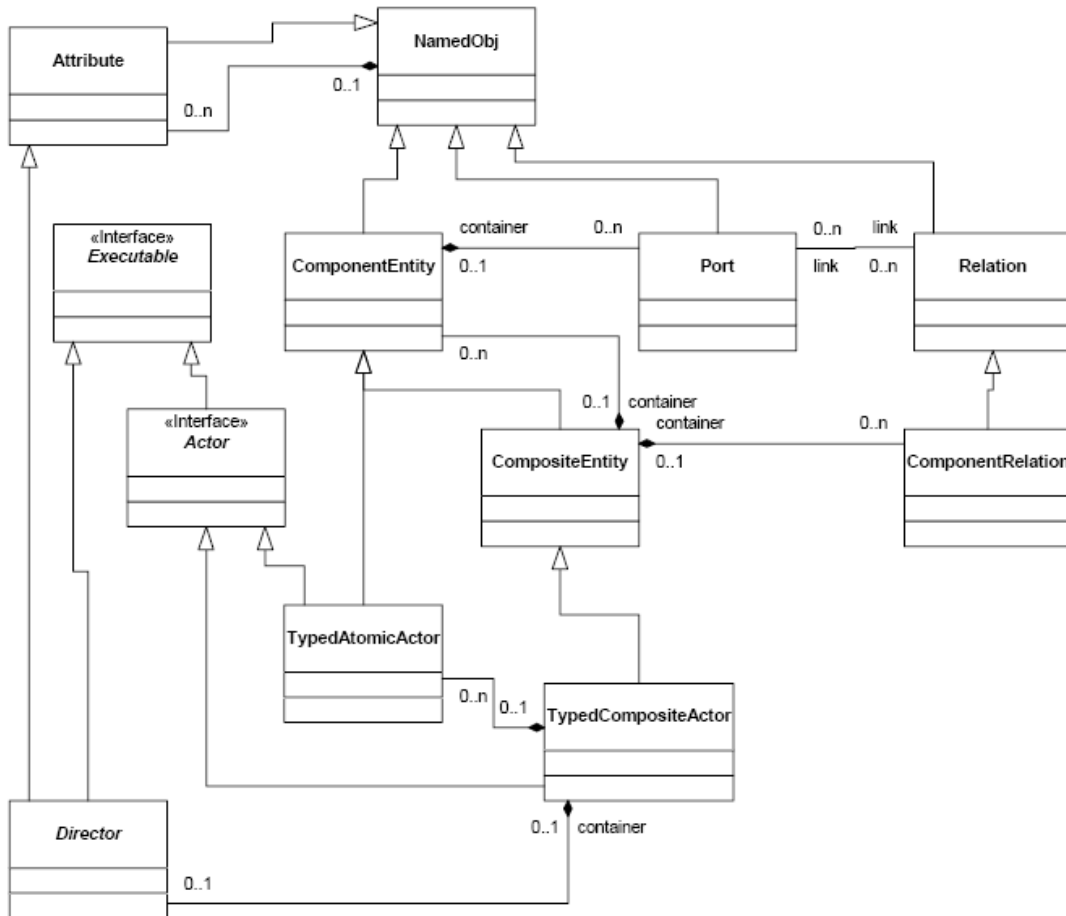
شکل (49) طراحی گیرنده برای مدل شکل قبل.

اگر چندین **property transformer** که قابل اجرا بر روی انتقال بخصوص هستند وجود دارد، سپس آنها در یک دسته اجرا می شوند. بنابراین عملیات که آنها روی مشخصات انجام می دهند باید مبادله پذیر باشد. آنها یک فیلد را انتخاب می کنند و آن را بوسیله یک ثابت افزایش می دهند.

مؤلفه های قابل اجرا واسط اکتور را پیاده سازی می

کلاس های کلیدی در Ptolemy II در شکل 50

کنند، می توانند اتمیک (*atomic*) یا مرکب (*composite*) اکتور های اتمیک در کد جاوا تعریف می شوند، در حالی که اکتور های مرکب از اکتور ها و رابط ها (تشکیل و اسمبل می شوند).



شکل (50) دیاگرام کلاس UML نشان دهنده کلاس های کلیدی در Ptolemy II

هر اکتور، خواه اتمیک و یا مرکب، شامل پورت هایی است که در یک اکتور مرکب از طریق رابط ها (*relation*) پیوند می یابند. اکتور ها، پورت ها و رابط ها همگی می توانند خاصیت هایی را دارا باشند (*parameter*). یکی از خصالت ها یک *director*. *director* یک نقش کلیدی در Ptolemy II بازی می کند.

آن معنای یک مرکب را تعریف می کند و مدل همزمانی و معانی انتقال را در Visualsense بیان می کند. *director* شبیه ساز را پیاده سازی می کند.

یک گره در شبکه بی سیم، اکتوری است.

توسعه ای که Visualsense پیاده می کند در شکل 51

که می تواند یک زیر کلاسی از هر یک از **TypedAtomicActor** یا **TypedCompositeActor**.

میان اینها است که برای **TypedAtomicActor**، رفتار در کد جاوا تعریف می شود در حالی که بر

اکتور ها که ارسال را به صورت بی سیم انجام می دهند پورت هایی دارند که مثال هایی از WirelessIOPort هستند. همچنین با هر پورت Ptolemy II, اکتور داده را بوسیله فراخوانی متد *send* یا *broadcast* روی پورت می

WirelessIOPort, *send*, *broadcast* نمی توانند پورت

بلوک دیگرام تصمیم بگیرند، زیرا چنین اتصالاتی وجود ندارد. در عوض آنها یک نمونه از کانال بی سیم را بوسیله نام تعیین می کنند و به آن نمونه محول می کنند تا مقصد () پیام ها را تصمیم گیری کند. نمونه بوسیله تنظیم پارامتر *outsideChannel* از پورت معادل با نام کانال بی سیم مشخص می شود (همه اکتور ها در یک سطح معین از سلسله مراتب نامهای یکسانی دارند. یک مشخصه بوسیله کلاس پایه تهیه می شود).

WirelessChannel و کلاس پایه *AtomicWirelessChannel* که در شکل ... نمایش داده شده

برای توسعه پذیری طراحی می شوند. *WirelessIOPort* با هم کار می کنند، که متد عمومی *transmit* برای ارسال داده ها استفاده می کند، که متد سه آرگومان می گیرد، یک *token*، یک *source port*، و یک *properties* ().

AtomicWirelessChannel یک دنباله ای از متد های محافظت شده (*protected*)، که در UML دیگرام بوسیله علامت راهنمای # نشان داده می شود. در روش کد گذاری Ptolemy II، ده نام هایی دارند که با علامت زیر خط () شروع می شوند.

این یک مثال ساده ای از الگوی استراتژی طراحی است که در آنجا کد رفتار در مقیاس بزرگ فراهم می کند و رفتار جزئی را به متد های محافظت شده محول می کند.

رفتار پیش فرض *AtomicWirelessChannel* بوسیله شبه کد زیر نمایش داده می شود:

```
Public void transmit(token,sender,properties) {
    Foreach receiver in range{
        _transmitTo(token,sender,receiver,properties)
    }
}
```

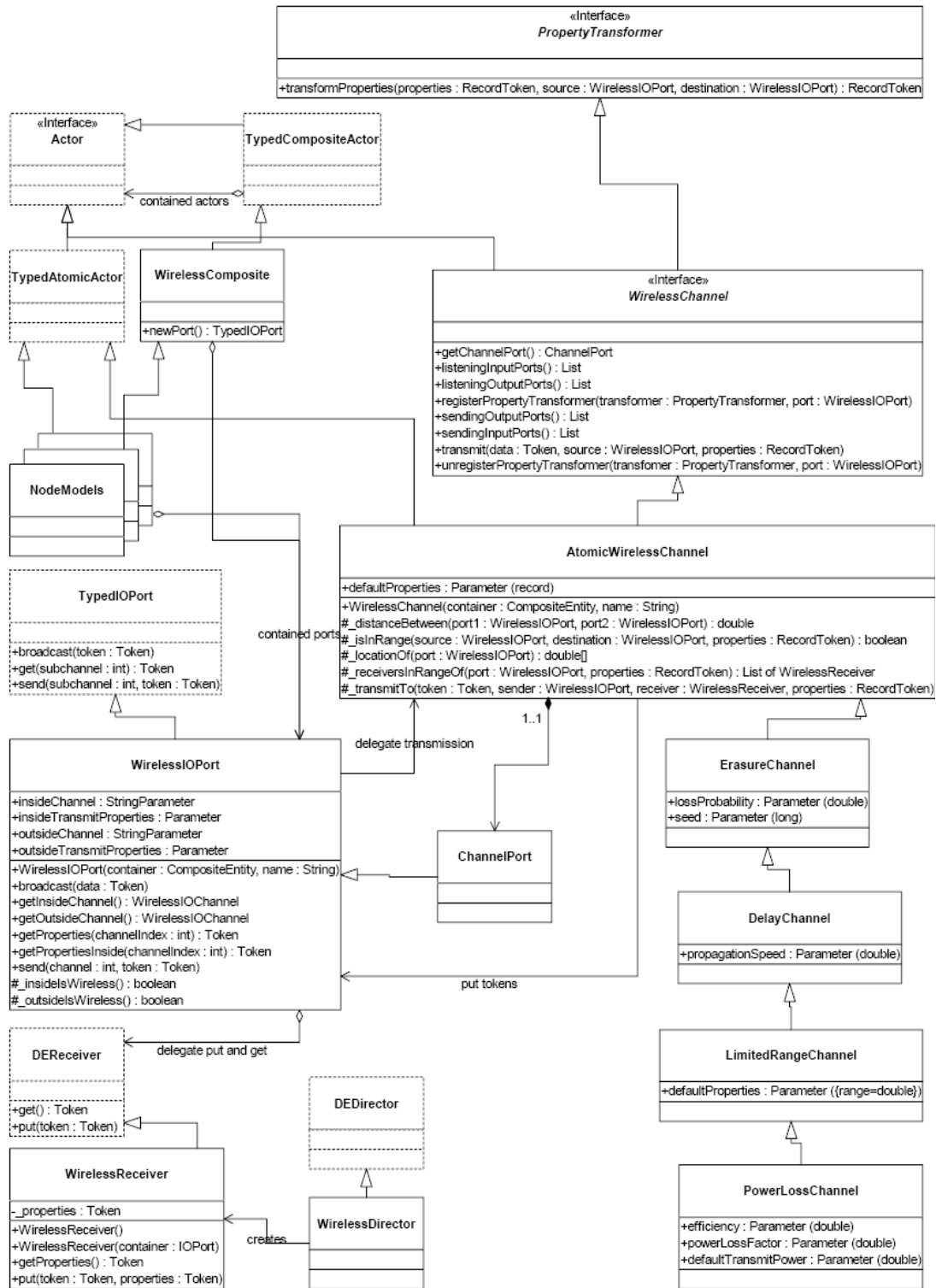
برای تصمیم گیری اینکه گیرنده در یک محدوده هستند، *receiverInRange()* را فراخوانی می کند که بوسیله پیش فرض همه گیرنده ها با همه پورت هایی که ارجاع می دهند به همان نام کانال، نگهداری می کنند، همچنین بوسیله فرستنده مشخص شد. *transmitTo()* پیش فرض متد عمومی

transformProperties() Properties استفاده می شود و سپس *token*

59

transformProperties() properties اصلاح شده را در داخل گیرنده قرار می دهند.

که استفاده از متد *registerPropertyTransformer()* را ثبت کرده است به کار می برد، ولی کار اضافه ای انجام نمی دهد. بنابراین اگر مبدل مشخصه ثبت نشده است پیش فرض *AtomicWirelessChannel* محدودیت های برد را ندارد و هیچ افت انتقالی را معرفی نمی کند. ما حالا می توانیم از طریق یک سری از مثال ها نشان دهیم که چه طور زیر کلاس ساخت مدل های مفصل تر (و مفید) را آسان می سازد.



شکل (51) دیاگرام UML نشان دهنده کلاس های کلیدی برای مدل سازی شبکه حسگر بی سیم

چند مثال و کاربرد:

در ادامه گزارش نمونه هایی از پیاده سازی و ایجاد گره های حسگر شرح داده شده است. در پتولومی چندین عامل موثر در آنالیز یک شبکه دخالت دارد مانند عمر باتری و رنج سیگنال و سیله ی مورد استفاده، همراه با در نظر گرفتن عواملی که می توانند اثر منفی در استفاده مفید از حسگرها داشته باشند. در این ابزار شبیه سازی شبکه هایی با گسترش قابلیت های حسگری نیز ممکن است. چنین قابلیت هایی شامل شناسایی زمین لرزه یا صدا و الگوریتم های مکان یابی همراه با بررسی داده در شبکه های حسگری است و بسیاری کاربرد ها که با افزایش آگاهی ما در مورد ارتباطات بی سیم ممکن خواهد شد. همان طور که قبلاً گفته شد شبکه های حسگر کاربردها و قابلیت های فراوانی دارند که به نظر می رسد این تنوع استفاده از . حال برای اینکه شبکه ای کارا تر داشته باشیم بایستی واکنش گره ها را کامل بفهمیم همراه با اینکه آنها چگونه با هم کار می کنند، همچنین مهم است که ما بدانیم: اولاً دلایلی که شبکه ی حسگر شکست می خورد و دوماً توانایی های توسعه یافته ی شبکه حسگری. در طراحی مدل شبکه های حسگر، فاکتور های مزبور باید به عنوان کارهایی که قبل از یک شبیه سازی صحیح، کامل شوند در نظر گرفته شود. کارهایی که باید انجام شود فهرست وار چنین است:

فهمیدن تعامل (واکنش) در شبکه های حسگر

مدیریت داده از طریق ارتباط بی سیم و کانال بی سیم و پیکربندی پورت ها

نقایص شبکه های حسگر

تأثیر از دست رفتن توان ، ، تأثیر بیرونی ، گره های حسگر در محیط های چند گانه

توانایی های توسعه یافته شبکه های حسگر

علاوه بر پردازش و حس کردن خصوصیتی نظیر لرزش ، ، حرکت و افزودن پردازش داده همراه با الگوریتم های ردگیری حرکت و صدا به مجرد کامل کردن این سه کار مهم مدل کردن یک شبیه سازی شبکه حسگر آغاز می شود.

طراحی و مدل کردن ناهمگن پتولومی

پروژه تحقیقی ناهمگن پتولومی، یک مدل کننده، شبیه ساز و طراح برای سیستم های همزمان با تمرکز روی سیستم های توکار با تکنولوژی ترکیبی است. مانند دستگاه های آنالوگ و دیجیتال، سیستم های نرم افزار سخت افزاری و وسایل الکترونیکی و مکانیکی. همچنین تمرکز روی سیستم هایی است که عمل حس کردن مختلط یا پیچیده دارند یعنی سیستم هایی که بسیاری عملیات مختلف از قبیل پردازش سیگنال، کنترل بازخورد، تصمیم گیری های متوالی و واسط های کاربری را ترکیب می کند. پتولومی توانایی مدل کردن سیستم های پیچیده را از طریق اکتور/مولفه مبتنی بر واسط کاربری یکی از اصول کلیدی پتولومی، استفاده از مدل های چند گانه محاسباتی در یک محیط طراحی ناهمگن مرتبه ای. این یک مزیت ویژه در دامنه ی حسگرها است. فاکتورهای متفاوت و فراوانی در اجرای یک شبکه حسگر. پتولومی شما را در بدام انداختن این فاکتورها به آسانی توانا ساخته است و باعث شده است که پتولومی یک ابزار همه کاره ی فراگیر شود.

مدل شبکه حسگر

شبکه های حسگر توانایی کنترل مناطق جغرافیایی چندگانه، و پردازش داده در شبکه را دارا می باشد. طراحی زیر بنایی از این شبکه های حسگر می تواند بسیار چالش برانگیز باشد. اساسی ترین هدف یک طراحی در پتولومی سه عامل اصلی مطرح شده در قسمت های قبل است. نقطه ی آغاز یک طراحی ایجاد جریانی از فرآیندها یا تکنیک طراحی قدم به قدم برای هر جزء شبکه های حسگر است.

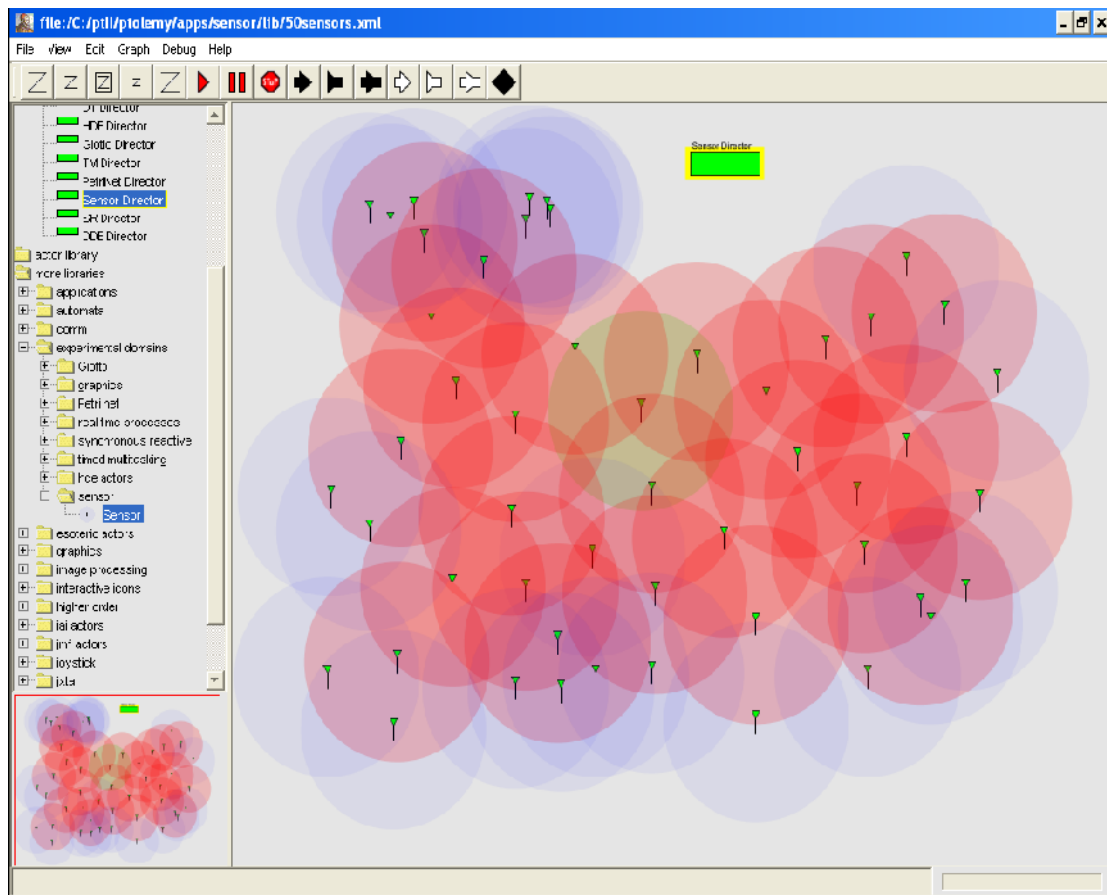
نمونه های ایجاد شده توسط نرم افزار:

(1) ⁶⁰:

یک ارزیابی از یک الگوریتم برای پرسش های رادیویی (broadcasting query) به طور مؤثر در شبکه حسگر شرح می شود. برای دستیابی به همه گره ها، موقعی که یک گره حسگر یک پیام رادیویی را دریافت می کند، آن ممکن است پیام را تکرار کند. مقصود کاهش تعداد چنین تکرار های مورد نیاز برای نفوذ کردن در شبکه است. یک تصویر از یکی از آزمایش های اجرا شده در شکل 52 در این تصویر، هر دایره یک گره حسگر را نمایش می دهد. در مرکز دایره یک آیکن برای نمایش یک آنتن است. این موقعیت فرستنده و گیرنده را برای گره نمایش می دهد. تصادفی پخش می شوند (یک اکتور در کتابخانه برای درک پخش تصادفی فراهم شده است). اجرای این مدل یک تعداد ثابتی از پرسش ⁶¹ ها را انتشار می دهد، که به عنوان یک پارامتر مد، از طریق شبکه حسگر در این آزمایش گره پایه با دایره رنگ سبز نیمه شفاف در شکل 52 نمایش داده شده است که یک پرسش را به طور متناوب منتشر می کند. در ابتدا همه گره ها این پرسش را به محض شنیدن آن تکرار می کنند. همچنانکه اجرا پیش می رود، توپولوژی شبکه اطراف آنها را یاد می گیرند و مبتنی بر الگوریتم توزیع شده ی شرح، تصمیم به ادامه تکرار یا توقف آن می گیرند. گره هایی که با آیکن قرمز رنگ در شکل 52، گره هایی هستند که تکرار می کنند در حالی که گره ها با آیکن آبی تکرار را انجام نمی دهند. بعد از آن الگوریتم همگرا شده است. در این آزمایش، گره های حسگر ارسال را از طریق یک کانال رادیویی بدون تأخیر زمانی و بدون هیچ نویز و اتلافی انجام می دهند. برد انتشار از گره حسگر یک پارامتری از گره است و با آیکن دایره ای نشان داده می شود. در شکل 52 گره ها یکسان است. به هر حال آن می تواند به آسانی تغییر داده شود.

⁶⁰ Flooding

⁶¹ Query



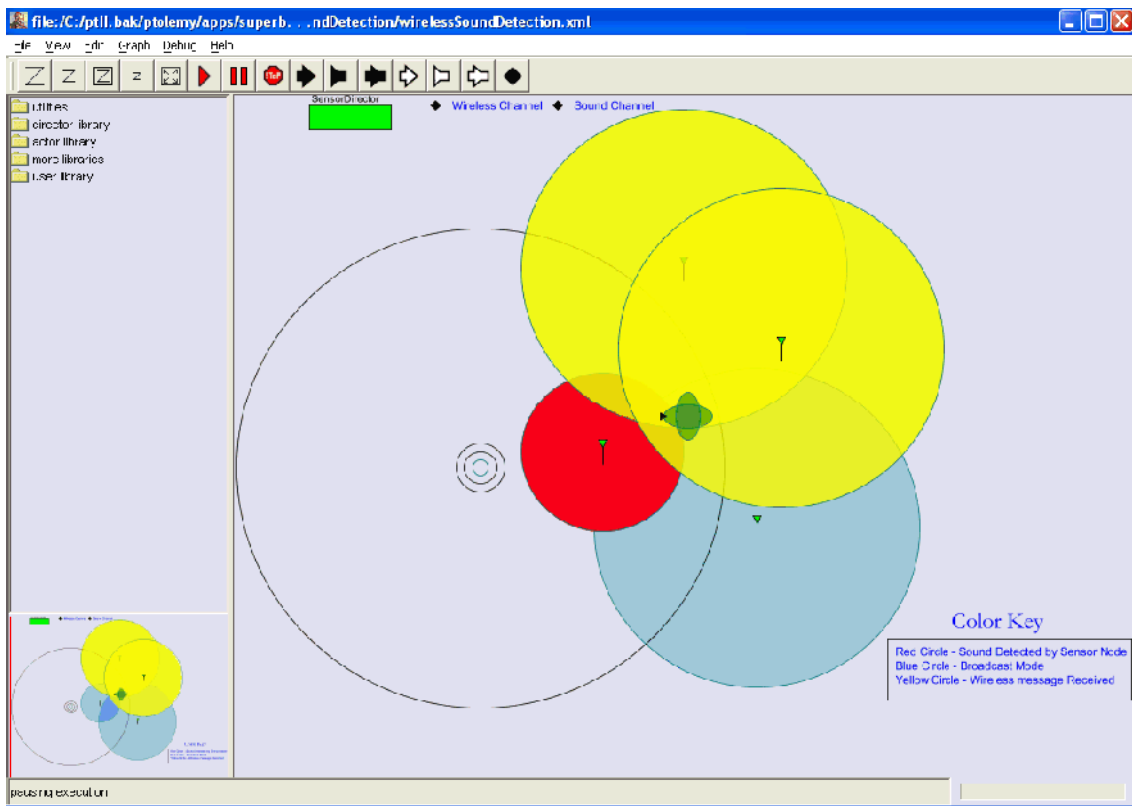
شکل (52) تصویری از مثال غرق سازی

(2) 62.

در این مثال از کاربرد یک مدل از منبع توان (در این مورد باتری ها) گره های حسگر با یک مدل از یک سناریوی که گره های صوتی با هم برای تصمیم گیری موقعیت منبع صوت بوسیله **triangulation** همکاری می کنند، تزویج می (شکل 53).

در این مثال یک کانال صوتی سیگنال های یک منبع متحرک را حمل می کند و گره های حسگر آن را از طریق یک کانال رادیویی انتقال می دهند. کانال صوتی تأخیر انتشار را در نظر می گیرد در حالی که کانال رادیویی به عنوان لحظه ای مدل می شود، شبیه به مدل مثال غرق سازی که در بالا شرح داده شد. آیکون های دایره ای گره ها برد ارسال خود را نمایش می دهند.

کانال صوتی سیگنال ها را از یک منبع صوت متحرک به گره های حسگر حمل می کنند. موقعی که یک گره حسگر یک سیگنال صوتی را کشف می کند، (زمان کشف و موقعیت گره حسگر) را از طریق کانال رادیویی منتشر می کند. یک اکتور ردیاب صوت مشاهدات را جمع آوری کرده زمان و موقعیت انتشار صوت از منبع متحرک را محاسبه می کند.



شکل (53) تصویر مثال مثلث بندی

3) نظارت بر ترافیک⁶³:

شبکه های حسگر بسیار دینامیک و پویا هستند. اشیا بوسیله رفت و آمد یک شبکه حسگر نظارت می شوند و ممکن است در اطراف یک میدان حسگر حرکت کنند. گره های حسگر جدید می توانند به شبکه ملحق شوند و موقعی که باتری یک گره حسگر تخلیه می شود آن شبکه را ترک می کند.

یک مدل طبیعی از چنین شبکه های حسگر پویایی باید از تغییرات در شبکه پشتیبانی کند نه فقط در توپولوژی اتصال داخلی، بلکه در مجموعه مؤلفه های موجود در شبکه بطوریکه قبلاً مشاهده شد، Ptolemy II برای تغییر در ساختار مدل پشتیبانی دارد. ما با مثال نشان می دهیم که چه طور این مدل سازی شبکه حسگر با کاربرد نظارت بر ترافیک به کار گرفته

حسگر ها در امتداد جاده برای جمع آوری اطلاعات که به پایگاه اصلی برای تحلیل بیشتر فرستاده می شود، توزیع می برای مدل سازی و شبیه سازی چنین شبکه ای ما ابتدا یک مدل برای میدان حسگر می سازیم، که شامل یک مؤلفه برای هر گره حسگر و کانال ها (یک کانال بی سیم برای انتقال میان حسگر ها، و یک کانال صوتی برای انتشار سیگنال از وسایل عبوری در حسگرها).

برای اینکه رفتار شبکه را شبیه سازی کنیم علاوه بر این به یک مدل محرک برای تولید ورودی ترافیک به میدان حسگر نیاز داریم. سؤال این است که چه نوع از ورودی باید برای مدل محرک در میدان حسگر فراهم گردد؟ به طور مشخص آن یک ماشین است که در امتداد جاده در یک کران و حاشیه وارد می شود. در این جهت، محرک واقعاً یک مدل ماشین (تومبیل) اضافه شده به میدان حسگر است. از آنجایی که ماشین ها می توانند در هر زمان وارد یک منطقه شده و پس

از مدتی آن را ترک کنند، میسر نخواهد بود تا آنها به طور ایستا در میدان حسگر مدل شوند. دینامیک به علت ورود و خروج ماشین ها، ما یک اکتور سطح بالا⁶⁴ را استفاده می کنیم (یکی که به عنوان مدل می گیرد بگر را). یک چنین اکتوری مدل دیگری که را که محاسبه اش را معین می کند، در بر می گیرد و در طول اجرا مدل⁶⁵ می تواند به طور پویا تغییر داده شود. اکتور سطح بالا دو ورودی دارد، با اولین ورودی داده را دریافت می کند که مدل نگهداشته شده باید پردازش شود و ورودی دوم تغییرات مدل را در مدل نگهداشته جاری دریافت می کند. تغییرات مدل می تواند مؤلفه های جدیدی از قبیل اکتورهایی که ماشین ها را مدل می کنند اضافه کند. موجود و اتصالات را حذف یا اضافه کنند.

موقعی که اجرا شروع می شود اکتور سطح بالا یک مدل درون تهی د. آن ابتدا یک تغییر مدل را برای ساخت میدان حسگر دریافت می کند، بعد از تغییر به کار برده شده، اجرا با هیچ وسیله نقلیه ای ادامه پیدا نمی کند. هنگامی که مدل ترافیک تصمیم گیری می کند که ورود ماشین به میدان به وجود آید، آن یک مدل ماشین را تولید می کند و به اکتور بالا آن را می فرستد، سپس مدل نگهداشته شده را برای قرار دادن مدل ماشین تغییر می دهد. اجرا با حرکت ماشین در منطقه بر طبق برنامه رانندگی ادامه می یابد و حسگر در مسیرش بوسیله کانال صوتی کشف می کند که آیا یک عبور اتومبیل وجود دارد یا خیر. اگر یک ماشین را شناسایی کند، حسگر سپس داده ای را به پایگاه اصلی می فرستد.

4) گمشده جنگی در منطقه دشمن و تعقیب کننده⁶⁶ :

در این مثال یک گره بی سیم گمشده جنگی⁶⁷ را در میدان حسگر ها شناسایی می کند (شکل 54)، حسگر ها در میدان با یکدیگر ارتباط برقرار می کنند تا یک درخت پوشا⁶⁸ از گره ریش (جایی که گمشده جنگی به آن نزدیک می شود) هایی که یک جهش⁶⁹ از گره ریشه و مانند آن هستند،. موقعی که حسگر گمشده جنگی را شناسایی می کند، به عنوان ریشه قرار می دهد و یک پیامی را به گره های همسایه اش پخش می کند. پیام شامل زمانی است که گمشده جنگی شناسایی می شود، موقعیت و عمق که برای گره ریشه از حسگر ها صفر است. کنترل می کند که آیا گره ریشه تغییر داده شده از آخرین بار (بوسیله کنترل زمان شناسایی آن) یا که یک مسیر کوتاه تر به گره ریشه وجود دارد. اگر چنین است، آن زمان شناسایی شده را ثبت می کند و اطلاعات⁷⁰ را به روز رسانی می کند (موقعیت و عمق در درخت) و یک پیام را که شامل زمان شناسایی، موقعیت و عمق در درخت است، به گره های همسایه آن ارسال می کند. با انجام این کار در شبکه حسگر یک درخت پوشا به طور توزیع شده ساخته می شود و حسگر ها بر طبق فاصله ای که از گره ریشه دارند، اندیس گذاری می شوند. با حرکت گمشده جنگی آن ممکن است بوسیله حسگر های دیگر شناسایی شود و درخت به طور پویا تغییر کند.

⁶⁴ Higher-Order

⁶⁵ contained model

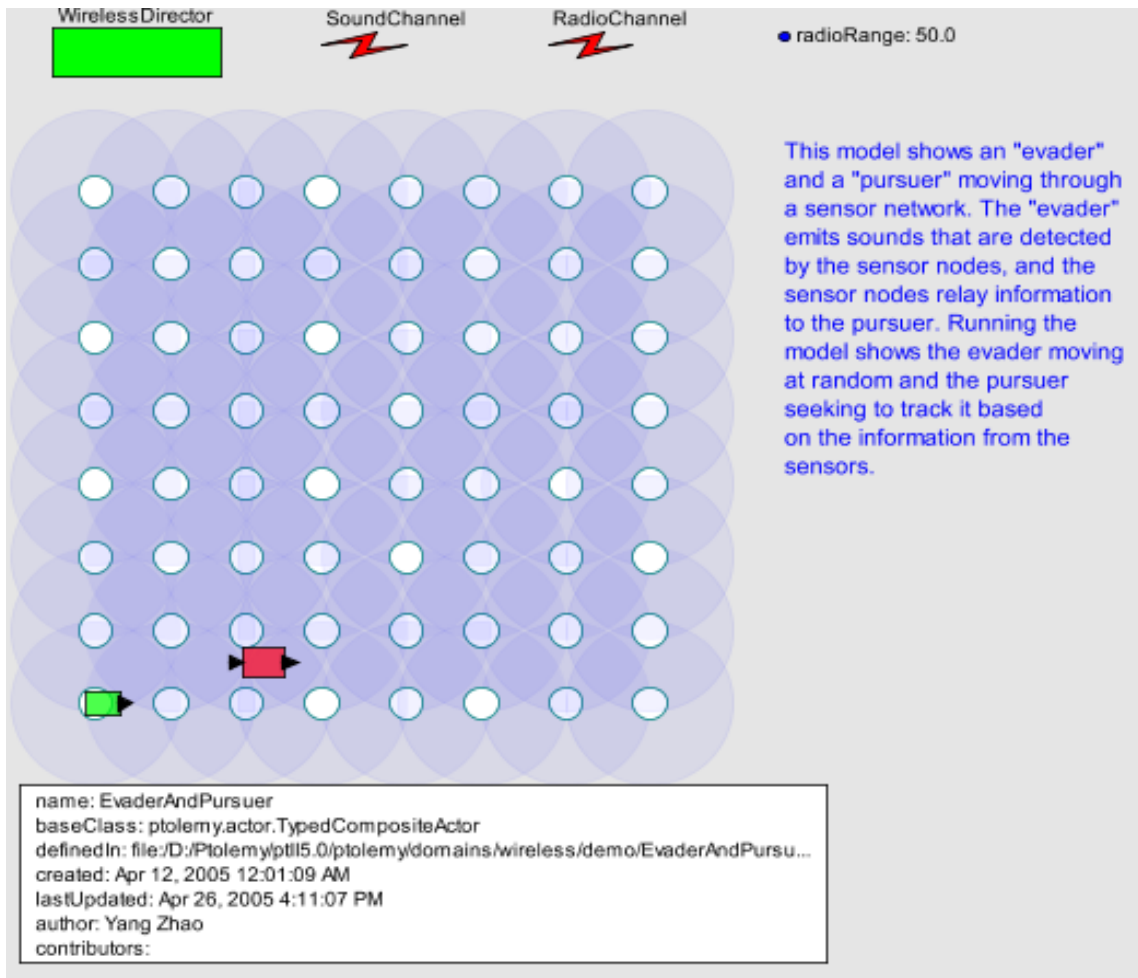
⁶⁶ Evader & pursuer

⁶⁷ Evader

⁶⁸ Spanning Tree

⁶⁹ Hop

⁷⁰ Parent



شکل (54) تصویری که میدان حسگرها را به همراه کانالها و... نمایش می دهد

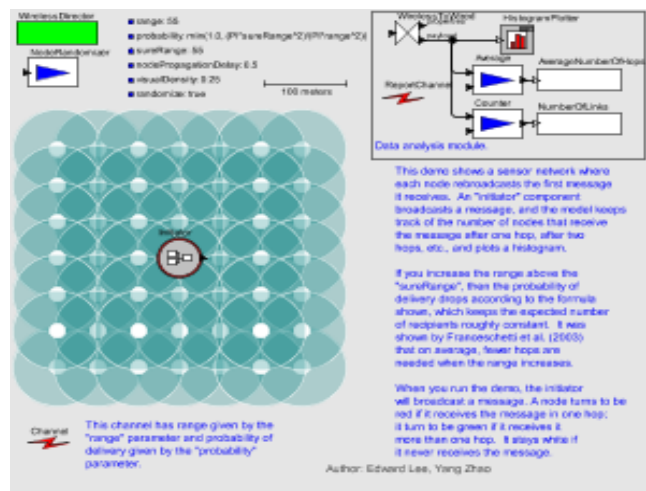
آیکون قرمز رنگ داخل میان حسگرها تعقیب کننده است و آیکون سبز رنگ گمشده جنگی است. موقعی که یک گره حسگر نشانه ای از پورت سیگنال (که برای دریافت سیگنال از تعقیب کننده یا گمشده جنگی استفاده می شود) را دریافت می کند، آن تشخیص می دهد که چه کسی سیگنال را فرستاده (با بررسی ابتدای سیگنال⁷¹). گمشده جنگی ارسال شده، گره خودش را به عنوان گره ریشه بعدی می نشاند و یک پیام را برای به روز رسانی درخت انتشار می دهد و گرنه آن یک پیامی به تعقیب کننده ارسال می کند تا آن موقعیت گره منشأ یا گره پدر را تشخیص دهد و تعقیب کننده با این اطلاعات به سمت گمشده جنگی حرکت خواهد کرد. موقعی که آن نشانه را از پورت ورودی که برای دریافت پیام از دیگر حسگرها استفاده می شود، دریافت می کند، بررسی می کند که آیا گره ریشه⁷² تغییر کرده یا یک مسیر کوتاه تر وجود دارد. اگر چنین است آن به روز رسانی را انجام می دهد و یک پیام را انتشار می دهد، در غیر این صورت پیام را مصرف می کند. تعقیب کننده از گره آخر درخت پوشا به طرف گره پدرش برای ردیابی گمشده جنگی حرکت می کند.

⁷¹ Signal header

⁷² root node

5) جهان کوچک⁷³:

این مثال یک شبکه حسگری را نشان می دهد که هر گره مجدداً اولین پیامی را که دریافت می کند انتشار می (شکل 55). یک مؤلفه به نام آغازگر⁷⁴ یک پیام را منتشر می کند، و مدل رد تعداد گره هایی را که پیام را بعد از یک بعد از دو جهش و همینطور الی آخر دریافت می کنند، نگهداری می کند و یک نمودار میله ای رسم می کند.



شکل (55) تصویری از small world

موقعی که مدل را اجرا می کنید، آغازگر یک پیام را منتشر خواهد کرد. یک گره زمانی قرمز خواهد شد که پیام را در اولین جهش دریافت کند، و در صورتی که پیام را بیش از یک جهش دریافت کند گره سبز می شود و اگر هیچ پیامی را دریافت نکند سفید خواهد ماند.

پارس بوک، مرجع دانلود کتاب الکترونیکی به زبان پارسی

www.parsbook.org

⁷³ Small world

⁷⁴ Initiator