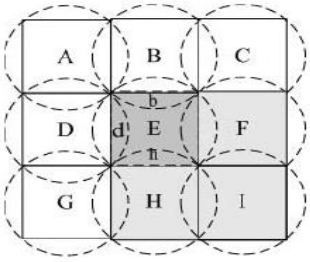
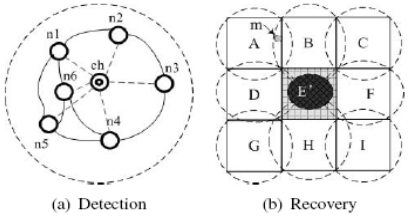
انرژی کارآمد مکانیزم تحمل‌پذیر خطا برای شبکه‌های بی‌سیم سنسور

ایده ا صلی مقاله: خو شه‌بندی یک توپولوژی کنترل کارآمد و پروتوکل ارتباط در شبکه‌هایی سن سور می‌با شد. با وجود اینکه سنسورها در محیطهای ناامن مثل میدان جنگ و جنگل و... گسترده شده‌اند و محدودیتهای از جمله محدودیت منابع و حجم‌کاری نامتوازن بین گره‌ها باعث شد که خوشه آسیب‌پذیر بوده و دارای خطا و اشتباه در ارتباط باشد. بنابراین نیاز شدید جهت بهبود قدرت مکانیزمهای تحمل‌پذیر خطای در کاربردهای واقعی شبکه‌های سن سور ا ست، در این مقاله یک مکانیزم توزیع شده تحمل‌پذیر خطا برای شبکه‌های سن سور پی شنهاد کردیم که CMATO (مکانیزم تحمل‌پذیر خطا بر پایه عضو خوشه) نامیده شده است در این دیدگاه خوشه، همچون یک شخص کامل برای نمایش نیازمندی‌های خو شه های دیگر و برای ک شف ترمیم خطاها با یک راه سریع و با انرژی کارآمد به کار می‌رود. این مکانیزم انعطاف‌پذیر برای متحد کردن طرحهای خو شه‌بندی گوناگون موجود در شبکه‌های سن سور می شود. علاوه بر این CMATO قادر به ترمیم برخی خرابی‌های سرخو شه‌ها نیز می‌با شد. بنابراین درCMATO ترمیم خرابی‌های گره‌ها، سرخوشه‌های چندتای و خرابی‌های اتصالات در خوشه، بهبود هر چه بیشتر کارائی و قدرت تحمل‌پذیر خطا در شبکه‌های سن سور کاربرد دارد. شبیه سازی نتیجه ن شان می‌دهد که این مکانیزم کارائی بالایی برای ترمیم خرابی سرخوشه و تحمل‌پذیر بودن خطا و مصرف بهینه انرژی نسبت به مکانیزم‌های موجود دارد.

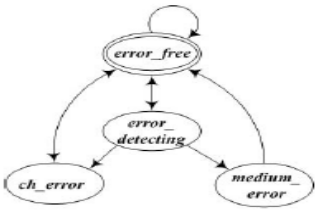
آنالیز مکانیزم: در شکل1 نشان داده شده وقتی که سر خوشه E خراب شود کل خوشه در یک خرابی است و منجر به ناتوانی فر ستنده‌هایش برای ار سال داده‌ها به بیرون خواهد شد. حتی بدتر زمانی ا ست که بسته‌های کوچک تولید شده توسط F و H و I از طریق E رله شوند. این خوشه‌ها نیز در یک وضعیت خرابی خواهند افتاد.



شکل 1



شکل 2



شکل 3

در مکانیزم CMATO گره‌های داخل خو شه موقع پردازش ار سالهای سرخو شه‌های هم سایه را ا ستراق سمع میکند. موقعیکه سرخو شه خراب می شودو ارتباط از سرخو شه قطع شود، اع ضاء خو شه آن را ک شف می‌کنند، باید بتواند خودشان را به خوشه مجاور برای ترمیم انتقال دهد.

شکل2(a) گره‌ها در داخل یک خوشه به سرخوشه خودشانch و همسایه این خوشه استراق سمع می‌کنند. بنابراین اعضاء خو شه می‌توانند خرابی سر خو شه را سریع ک شف کنند. در شکل2(b) وقتی همه گره‌ها از خرابی سرخو شه E اطلاع پیدا می کنند بعضی از گره‌ها (گره‌های واقع در حوزه خاکستری رو شن) به خو شه همسایه متصل می شوند در حالی که دیگران (گره‌های واقع در حوزه خاکستری تاریک) به خوشه E که اخیراً ساخته شده متصل می‌شوند از این گذشته4حالت برای گره‌ها هستند:1ـ بدون خطا (Error-free)2ـکشف خطا(error-detecting) 3ـخطای سرخوشه -ch) (error4ـ خطای میانی(medium-error) که در شکل3 مشاهده می‌کنید. انتقال از حالت کشف خطا به بدون خطا، خطای ch، یا خطای میانی عبارت کشف خطا است. زمانی که از خطای سرخوشه یا خطای میانی به بدون خطا می‌رویم عبارت ترمیم خطا هست.

A ـــــ کشف خطا: کشف خطا اولین گام در مکانیزم تحمل‌پذیر خطا می‌باشد ما کلاً در نظر می‌گیریم که خوشه خطاهای خودش را می‌تواند کشف کند. موارد زیادی هست که اعضاء خوشه می‌توانند از وضعیت سر خوشه‌ها اطلاع پیدا کنند. برای مثال وقتی که اعضاء خوشه در حال ارسال است سرخوشه داده را حس می‌کند.

در مکانیزم CMATO ، یک گره ارسالهای سرخوشه همسایه را وقتی که خوشه آن در حالت بدون خطا هست استراق سمع می‌کند. اگر نتواند در هیچ فریم کوتاه یا نمایش بسته‌ای کوتاه زندگی سرخوشه آن را استراق سمع کند. پس می‌تواند خطای پیدا کند. جهت انتقال به حالت کشـــف خطا باید دریافت‌کننده‌اش را باز نگهدارند، آن وقت این گره مجبور است تصمیم بگیرد آیا خرابی سرخوشه یا خطای میانی علت شکستن ارتباط به سرخوشه شده؟

درCMATO یک مکانیزم مشاوره در خوشه برای طبقه‌بندی صحیح نوع خطاها استفاده می‌کنیم. اگر بیشتر از α درصد گره‌ها در خوشه برای کشف خرابی ارتباط به سرخوشه باشد. پس سر خوشه باید خراب اعلان شود. حتی اگر سرخوشه خراب نشده باشد. اما بیشتر اعضاء خوشه نتوانند با سرخوشه‌های آنها ارتباط برقرار کنند به این خاطر خطای میانی هست پس ما می‌توانیم به این علت خطا را خرابی سرخوشه طبقه‌بندی کنیم، در این صورت دلیل خوبی برای انتخاب سر خوشه جدید می‌باشد. CMATO از لیست غیر فعال برای ارائه دوباره لیست اعضای خوشه که از سرخوشه قطع ارتباط کردند استفاده می‌کند بنابراین لیست غیر فعال انتشار و بروزرسانی میان اعضاء سرخوشه می شود. اگر طول لیست بیشتر از (α × |C| |C تعدادی از گره‌های عضو در خوشه C) باشد پس سرخوشه خراب اعلام می‌شود.

خرابی سرخوشه توسط انتشار پیام ch-fail اعلام میشود. گرهی که این پیام را برای اولین بار دریافت می‌کند خواهد توانست آن را فوراً رله بکند تا اینکه تمام اعضای خوشه بتوانند بدانند که سرخوشه خراب شده است اعضاء خوشه که پیام ch-fail را دریافت کردند و می‌توانند سریعاً وارد فاز ترمیم شوند. بعد از زمانی فاصله اگر هیچ پیامی ch-fail انتشار نیافته باشد. پس قطع ارتباط به سرخوشه‌اش به علت خطای میانی می باشد.

B ـ ترمیم خطاها: برای ترمیم خطا هر گره یک مجموعه سرخوشه همسایه (NCH) نگهداری می‌کند. در CMATO وقتی که سرخوشه‌ها با همدیگر اتصال پیدا می‌کنند اعضاء سرخوشه به راحتی می‌توانند سرخوشه‌های مجاور را به NCHSشان اضافه کنند. برای مثال در شکل1 گره‌هایی که در ناحیه b و f و h و d قرار دارند.

درخوشـــه E با یدســـرخوشـــه B وF وH وD را درNCHS به ترتیب اضـــافه کنندوقتی که اعضـــاء خوشـــه در ناحیه‌های{b,f,h,d} سرخوشه E را پیدا کنند یا خراب یا اتصال به سرخوشه شکسته یا قطع شده است. آنها می‌توانند خودشان را به خوشه مجاور انتقال دهند بهرحالNCH گره‌ها در ناحیه{(E–(b+f+h+d} قرار می‌گیرند باید خالی شود. بنابراین سرخوشه‌های جدید نیاز به انتخاب جهت ترمیم این گره‌ها دارند.

1) ترمیم خطای میانی: این خطا اساس برای خرابی‌های سرخوشه‌ها می‌باشند. اگر عضو خوشه خطای میانی را کشف کند برای اتصال به خوشه‌های همسایه تلاش می کند. اگر مجموعه سرخوشه‌های همسایه خالی نباشد(|NCH|≥1). آن فقط پیام درخواست اتصال را به بهترین سرخوشه همسایه می‌فرستد. با دریافت این پیام ، سرخوشه مجاور می‌تواند پیام تصـــدیق این پیام و برای دریافت داده از این گره آماده کند. بنابراین این گره ترمیم شـــده اســـت. اگر مجموعه سرخوشه همسایه خالی باشد(|NCH|=0). پس عضو خوشه‌ای برای انتخاب یک گره از همسایه‌هایش همچون گره رله هست از میان این عضو خوشه میتواند داده را به سرخوشه ارسال کند. ترمیم سرخوشه مجاور بهتر از رله گره می‌باشد . زیرا بیدار ماندن باعث ساختن گره رله می شود. که آن گره سرخوشه بوده است. این روش خواهد تواند تعداد سرخوشه‌ها را افزایش دهد. و تداخل میان خوشه‌ها را افزایش دهد. وقتی که چندین خطای میانی در شبکه اتفاق می‌افتد.

2 ــ ترمیم خطای سرخوشه(ch-error) :با خرابی سرخوشه همهٔ عضوهای خوشه توسط انتشار پیام ch- fail اطلاع پیدا می‌کنند. اعضای خوشه می‌تواند برای سرخوشه شدن با تابع وزن رقابت کند. $f(c_{im}) = r * \frac{|INbr|}{C_i} + (1-r) * \frac{E_{cur}(c_{im})}{E_{max}(c_i)}$

که c_{im} عضو خوشه m^{th} خوشه C_i هست، |INbr| اندازه مجموع همسایه در خوشه است، |Ci| اندازه خوشه است. بنابراین $\left(\frac{|INbr|}{Ci}\right)$عامل اتصال را نمایش می‌دهد. $E_{cur(cim)}$ انرژی باقیمانده از آن گره، $E_{max(ci)}$ یک برآورد از ماکزیمم انرژی باقیمانده میان گره‌ها در خوشه C_i هست $\left(\frac{Ecur(cim)}{Emax(ci)}\right)$ بنابراین معیار انرژی را نمایش می‌دهد .

بر طبق تابع وزن f وقتی یک گره تصمیم به سرخوشه شدن می‌گیرد یک پیام آگهی سرخوشه ch-adv انتشار می‌دهد. با دریافت این پیام گره‌ها می‌توانند تایمرشان از سرخوشه‌های مناسب قطع کنند. موقعی که سرخوشه جدید انتخاب شد عضوهای سرخوشه خراب باید یا NCH شان بروز شوند. و پیام درخواست اتصال به بهترین سرخوشه‌ها در INCHSارسال کنندیااگرNCHخالی باشد به گره‌های مجاور رله میکند.

مزایا:مکانیزمCMATO برای ترمیم خطا های دائمی در سمت سرخوشه بخوبی خطاهای میانی، بین سرخوشه واعضای خوشه عمل می‌کند.

ایستگاه اصلی نیاز به دانش عمومی و پیش برآورد، دربارهٔ موقعیت گره‌ها بوسیله پردازش متمرکز ندارد.

کشف و ترمیم هر دو به طور موضعی در زمان اجرا به کار می‌افتند. شبیه‌سازی نتیجه نشان می‌دهد که CMATOمیتواند در داخل الگوریتم‌های خوشه‌بندی موجود قرارگرفته برای کشف خطا در گره‌ها و سپس ترمیم دینامیکی شبکه از خطا با انرژی کارآمد به کار رود.

معایب: وقتی که چندین خطای میانی در شبکه اتفاق می‌افتد؛ برای ترمیم بیدار ماندن باعث ساختن گره رله می‌شود و تعداد سرخوشه‌ها و تداخل میان آنها را افزایش می دهد.

پارامترهای به کار رفته در این مکانیزم: این مکانیزم در زمان اجرا اقدام به کشف و ترمیم می کند پس پارامتر در دسترس بودن تحمل پذیرخطا را تحت تأثیر قرار می دهد.

شبیه سازی: CMATO را در محیط J-sim پیاده‌سازی شده. در این آزمایشات صد گره به طور تصادفی در محیطی به ابعاد200×200مترمربع گسترش دادیم که گره sink در مکان (0,0) قرار داده شده است. شبکه با استفاده از LEACH وHEED خوشه‌بندی شده سرخوشه‌ها در درخت پوشا برای مسیریابی سازمان یافته‌اند. سرعت انتخاب سر خوشه 0.2 است. و محدوده خوشه 50متر می‌باشد. برای تست رفتار شبکه خطاها را به شبکه تزریق می‌کنیم. برای هر سرخوشه اگر K^{th} خطا در زمان T_k وارد شود. سپس زمان ورود به صورت X1 = T,Xk=Tk–Tk-1 , for K=2,3,… تعریف می‌شود.

فرض کنیدXi مستقل است به طور همانندی متغیرتصادفی توزیع شده است وبه توزیع‌نمایی کلاسیک با نرخ λ تعلق دارد$f_{Ti}(t)= \lambda e^{-\lambda t}, t>0$

طبق منبع [14] جریان زمان ورودی {x1,x2,x3,…} در واقع شکل فرآیند پواسون است λ مساوی E(xi) که مقدار پیش‌بینی شده x_i است. ظاهراً بزرگتر از λ مکرراً خطا اتفاق می‌افتد. بنابراین عضو خوشه زمانی که حجم کاریش کوچکتر از سرخوشه است. ما λ اش را 10 زمان مساوی با λ سر خوشه قرار می‌دهیم. وقتی خطای سرخوشه اتفاق می‌افتاد فرض می‌کنیم خرابی ابدی است و سرخوشه از هر کدام از اتصالها به گره‌های دیگر قطع شده است. وقتی خطای میانی اتفاق می‌افتاد در عضو سرخوشه اتصال از عضو خوشه به سرخوشه قطع شده است. همه خرابی‌ها گره‌ها فعالند و همه اتصالات شکسته شده دوباره اتصال می‌یابند. بعد از آخرین خطا برای rts × 6.2 که rts چرخش زمان محدود را مشخص می‌کند.

مورد مقایســـه:CMATOرا در هر دو پروتوکلHEEDوLEACH پیاده‌ســـازی کردیم با تولید خطا در شـــکل4 کارائی CMATO با CHATOوقتی λ = 800 و rts=2000s است مقایسه کردیم.CMATO می‌تواند خرابی‌های سرخوشه را کشف و ترمیم کند در حالی که 87% خطاها در CHATO کشف می‌شود علاوه بر این CMATO قادر به ترمیم همه خطاهای میانی اســـت. در حالی‌که CHATO هیچ مکانیزم برای ترمیم خطا ندارد بنابراین در شـــکل 5 ما تنها خطاهای خرابی سرخوشه را جهت مقایسه بیشتر کارایشان تزریق کردیم.

TABLE I
PARAMETERS AND THEIR VALUES

Parameter	Value
Electronics energy (E_{elec})	50 nJ/bit
Threshold distance (d_0)	87 m
Transmit Amplifier ($\epsilon_{fs}, \epsilon_{mp}$)	$10.00pJ/bit/m^2$, $0.0013pJ/bit/m^4$
Data packet size	4000 bytes
Beacon packet size	20 bytes
Initial energy	1.0 J/battery

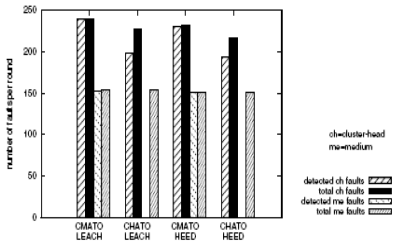


Fig. 4. Faults detected vs. total faults. ($\lambda = 800, rts = 2000$)

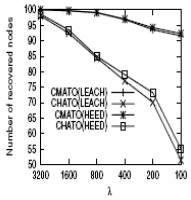


Fig. 5. Impact of λ to the number of nodes recovered from cluster head failures. $rts = 2000$

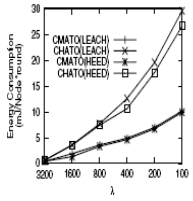


Fig. 6. Impact of λ to the energy consumption. $rts = 2000$

λنرخ پوا سون را برای مقای سه تغییر می‌دهیم. وقتی که خطاها کم ه ستند(λ≥3200) شبکه تو سط هر دو روش ترمیم می شود. و وقتی‌که به طور مکرر بیشتر می شود بیشتر خطای سرخو شه‌ها در مکانیزمCHATOترمیم نشده‌اند. به این دلیلCHATO تنها می‌تواند مقدار خطای منفرد را کشـــف کند. وقتی چندین خطا در همان زمان اتفاق بیفتاد به احتمال زیاد خوشه همسایه درCHATO نمی‌تواند کار بکند.

هزینه مکانیزم تحمل‌پذیر خطا شـــامل اســـتراق ســـمع تغییرات کنترل پیام می‌باشـــد. در شـــکل 6 هزینه انرژی CMATO را می‌توانید ببینید. تقریباً خطی با نرخ λ ا ست. در صورتی‌که برای مکانیزم CHATO وقتی خطاهای بی شتر اتفاق بیافتد و سرخوشه‌های بیشتری خراب می شوند، سرخوشه‌های جدید درست نمی‌شود. بنابراین سرخوشه مبنی بر مکانیزم تحمل‌پذیر خطا مجبور هستند محدوده ارتباط‌شان برای نمایش ارتباط و ارتباطات داخلی خو شه گسترش یابند. بنابراین خط شیب CHATO عمیق‌تر از CMATO ا ست. انرژی م صرفی CMATO بی شتر از 60 در صد کمتر از CHATO وقتی که λ≤ 400 است.

کارهای آتی: ندارد.