Packet switching

Ir. J. P. Katoen
Ir. J. van de Lagemaat

1 Inleiding

In de jaren zestig werd de behoefte om gegevensverwerkende apparatuur te koppelen en informatie daartussen uit te wisselen steeds groter. In eerste instantie gebruikte men de beschikbare circuitgeschakelde telefoonnetwerken voor datacommunicatie, bijvoorbeeld door een koppeling van apparatuur via een modem met een telefoonaansluiting. De aanvoer van de te versturen data staat echter in het algemeen niet garant voor een continue invoerstroom, maar heeft een nogal 'bursty' karakter. Daarom is het gebruik van circuit switching ronduit ongunstig – systeemelementen moeten onnodig a priori worden geclaimd. Deze inefficiëntie heeft geleid tot de introductie van *packet switching* (ofwel: pakketschakelen).

Bij packet switching worden systeemelementen alleen geclaimd wanneer er informatie moet worden verstuurd, dus als er echt behoefte aan is. Alleen al vanwege deze eigenschap wordt packet switching tegenwoordig toegepast in de meeste netwerken voor datacommunicatie.

Dit hoofdstuk is als volgt ingedeeld. In paragraaf 2 wordt het principe van packet switching uiteengezet. De toepassingen van packet switching komen aan de orde in paragraaf 3. Voor de ondersteuning van packet switching zijn diverse functies in het netwerk nodig. Deze worden besproken in paragraaf 4. In paragraaf 5 gaan we in op de kwantitatieve aspecten van packet switching zoals de vertraging van een pakket en de efficiëntie van packet switching. We eindigen in paragraaf 6 met een overzicht van de standaardisatie die sterk gerelateerd is aan packet switching.

2 Principe

De naam packet switching komt voort uit het feit dat bij deze schakeltechniek de te transporteren informatie wordt uitgewisseld in de vorm van informatie-eenheden. De te transporteren informatie duiden we verder aan met de term *berichten*; in OSI-terminologie is een bericht equivalent met een *SDU* (Service Data Unit). De informatie-eenheden worden *pakketten* genoemd; in OSI-jargon is een pakket een *PDU* (Protocol Data Unit).

Hoewel pakketten doorgaans een vaste lengte hebben, zijn er ook netwerken die een variabele pakketlengte toestaan. Het idee is om de totale hoeveelheid informatie pakketsgewijs door het netwerk te transporteren. In een tussenknooppunt, bijvoor-

beeld een lokale centrale, wordt het pakket (kort) opgeslagen en vervolgens verder verstuurd zodra capaciteit vrij is op de uitgaande verbinding. Dit principe, bekend onder de term *store-and-forward*, wordt herhaald in elk tussenknooppunt, totdat de bestemming is bereikt. Het principe van packet switching is weergegeven in figuur 1. Hierbij wordt een bericht opgedeeld in drie pakketten van vaste lengte. De pakketten worden ontvangen en opgeslagen in een tussenknooppunt en vervolgens verder verstuurd. De vertraging die ieder pakket ondervindt, kan verschillend zijn.

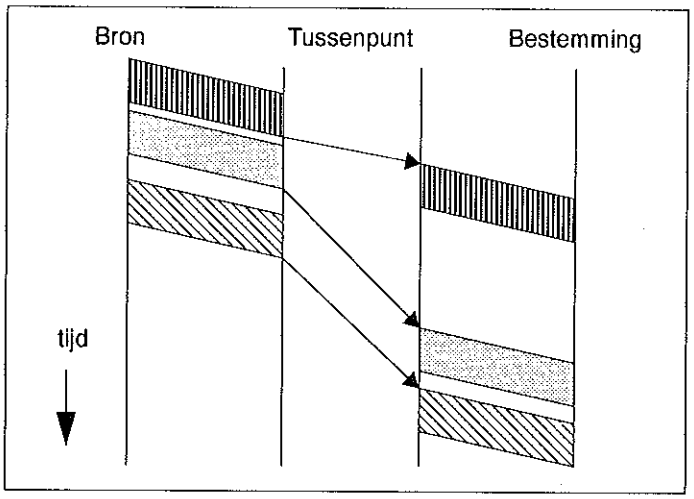

Fig. 1. Voorbeeld van packet switching.

Relatie met andere schakeltechnieken

Ten opzichte van andere schakeltechnieken (circuit switching, message switching) is packet switching het meest verwant met message switching. Bij message switching is de te transporteren eenheid een compleet bericht; bij packet switching is de eenheid het pakket, waarvan de grootte geen verband hoeft te houden met de berichtlengte. Als bij packet switching de pakketlengte kleiner is dan de berichtlengte, wordt het bericht in een aantal pakketten verstuurd. In dat geval kan een deel van het bericht al bij de bestemming zijn aangekomen, terwijl de rest van het bericht nog onderweg is of nog moet worden verstuurd. Bij packet switching kunnen pakketten onderling uit volgorde geraken doordat pakketten bijvoorbeeld via een verschillende route de bestemming bereiken. Hierop gaan we in paragraaf 4 nader in.

Circuit switching en packet switching verschillen in diverse aspecten.
Het belangrijkste onderscheid is wel dat bij circuit switching de middelen, zoals verbindingen en verwerkingscapaciteit in

knooppunten in het netwerk, van tevoren moeten worden gereserveerd. Bij packet switching hoeft er pas een verbindingsweg beschikbaar te zijn op het moment dat er werkelijk informatie moet worden verstuurd. Een direct gevolg hiervan is dat de kosten voor de gebruiker van packet switching evenredig zijn met de hoeveelheid informatie die wordt uitgewisseld. De kosten van circuit switching zijn niet direct afhankelijk van de verzonden hoeveelheid informatie, maar worden voornamelijk bepaald door de tijdsduur van de gebruikte verbinding.
Verder is een belangrijk verschil dat circuit switching niet flexibel is wat betreft het bandbreedtegebruik (zie II 2010, Circuit switching), terwijl bij packet switching op een efficiënte wijze verschillende bandbreedten aan de gebruiker kunnen worden aangeboden.

Logische verbindingen

Packet switching biedt de mogelijkheid tot het vormen van logische verbindingen die flexibel en dynamisch gebruik maken van de verschillende links en knooppunten in het netwerk. Pakketten voor een bepaalde logische verbinding kunnen over verschillende links worden verstuurd (men noemt dit *splitting*) en pakketten van verschillende verbindingen kunnen over dezelfde link worden verstuurd. Hierdoor kan de capaciteit van het netwerk beter worden benut. Figuur 2 toont multiplexing en splitting voor *twee* verbindingen (A en B), waarbij pakketten over *drie* verschillende links worden verstuurd. Aan de ontvangende kant moeten de pakketten voor iedere verbinding worden verzameld (*de-multiplexing*) en op volgorde worden gezet (*sequencing*), voordat ze aan de gebruiker van de verbinding worden doorgegeven.

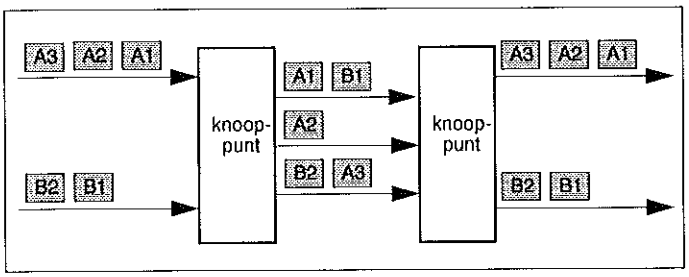

Fig. 2. Het vormen van logische verbindingen.

Gebruikersperspectief

De gebruiker van een netwerk heeft geen weet van de gebruikte schakeltechniek door het netwerk: de schakeltechniek is voor hem grotendeels transparant. Voor de gebruiker is alleen het verschil tussen *verbindingsloze* (connectionless) en *verbindingsgeoriënteerde* (connection-oriented) services van belang. Er is sprake van een verbindingsgeoriënteerde service als de gebruikers eerst een verbinding moeten opbouwen, vervolgens informatie kunnen uitwisselen, waarna de verbinding expliciet moet worden verbroken. Voor de gebruikers lijkt het of er een

'echte' fysieke verbinding aanwezig is; in werkelijkheid is er sprake van een logische verbinding door het netwerk. Bij verbindingsloze services hoeft er geen verbinding door de gebruiker te worden aangevraagd en verbroken. De nadruk ligt bij deze service op het efficiënt transporteren van een bericht. Circuit switching en packet switching kunnen beide worden gebruikt ter ondersteuning van zowel verbindingsloze- als verbindingsgeoriënteerde services (zie ook II 2010, Circuit switching).

3 Toepassingen

In de meeste openbare datanetwerken wordt packet switching toegepast. Deze schakeltechniek wordt vooral toegepast in netwerken die een niet al te grote snelheid hebben (typisch zo'n 64 kbit/s, bijvoorbeeld netwerken die zijn gebaseerd op de CCITT-aanbeveling *X.25*).

Vaak wordt een combinatie van circuit switching en packet switching gebruikt, waarbij een gedeelte van het netwerk is gebaseerd op de ene schakeltechniek en een ander deel op de andere schakeltechniek. Een voorbeeld hiervan is *ISDN*. ISDN biedt de gebruiker twee B-kanalen en één D-kanaal. Het D-kanaal is voornamelijk bedoeld voor de overdracht van signaleringsinformatie en is gebaseerd op packet switching. De B-kanalen, die voor de overdracht van gebruikersinformatie dienen, maken gebruik van circuit switching.

ATM

Door de complexe protocolondersteuning die voor packet switching noodzakelijk is, werd packet switching aanvankelijk niet gebruikt voor hoge-snelheidsnetwerken (in de orde van tientallen Mbit/s). Door de steeds toenemende kwaliteit van transmissiemiddelen (onder andere door toepassing van glasvezel) kunnen de protocollen echter sterk worden vereenvoudigd, waardoor hogere snelheden kunnen worden bereikt.

Een eerste aanzet tot een reductie in de benodigde protocolondersteuning heeft geleid tot de techniek *frame relay* (zie II 2023, Frame relay). Frame relay biedt daardoor bij het doorschakelen van pakketten in knooppunten hogere snelheden dan packet switching en is bijzonder geschikt voor de interconnectie van lokale netwerken.

Packet switching waarbij een minimum aan ondersteunende functies door het netwerk wordt geleverd, ligt ten grondslag aan het principe van *ATM* (Asynchronous Transfer Mode, zie II 2025, ATM). Dit principe wordt gebruikt in veel hoge-snelheidsnetwer-

ken en wordt door het CCITT aanbevolen voor B-ISDN (breedband-ISDN). ATM heeft de volgende karakteristieken [1]:
- ATM is gebaseerd op packet switching met pakketten van vaste lengte (53 octetten, waarvan 48 octetten gebruikersinformatie).
- Er vindt geen foutafhandeling en flow control plaats per verbinding (zoals in X.25), maar alleen tussen zender en bestemming (op OSI-Transportlaag-niveau).
- Er wordt verbindingsgeoriënteerd gewerkt.
- De hoeveelheid overhead-informatie in een pakket is beperkt.

Lokale netwerken

Circuit switching wordt niet gebruikt in *lokale netwerken* (Local Area Networks, LAN's). Een uitzondering hierop vormen de PABX'en. Packet switching is in LAN's de schakeltechniek bij uitstek. Standaarden zoals de IEEE 802.X-serie met onder andere specificaties van CSMA/CD (Ethernet), Token Ring en Token Bus zijn alle gebaseerd op packet switching en worden veelvuldig in bestaande netwerken toegepast.

Multimediacommunicatie

De behoefte om informatie van verschillend karakter te transporteren, neemt hand over hand toe. Transmissie van spraak, video, audio, enzovoort moet tegenwoordig als een geïntegreerd pakket aan de gebruiker worden aangeboden. Doordat multiplexing bij packet switching eenvoudig is, is deze schakeltechniek tamelijk flexibel bij het ondersteunen van diensten met verschillende capaciteitseisen. Dit is voor het CCITT één van de belangrijkste redenen geweest om ATM te kiezen als transporttechniek voor breedband-ISDN.

Mobiele netwerken

Packet switching wordt toegepast in de huidige *mobiele datanetwerken*. De eerste mobiele netwerken waren nog volledig gebaseerd op circuit switching, maar naarmate de behoefte groter wordt om ook data en gedigitaliseerde spraak via mobiele systemen over te dragen, wint het principe van packet switching terrein. Een voorbeeld hiervan is het Mobitex-systeem (oorspronkelijk ontwikkeld door Eritel AB en geëxploiteerd door onder andere RAM Mobile Data Inc.) dat zowel pakketgeschakelde datacommunicatie als circuitgeschakelde spraakverbindingen ondersteunt. Het Mobitex-systeem ondersteunt een maximale pakketlengte van 512 bytes en is in staat segmentatie (zie paragraaf 4.3) uit te voeren in het toegangsnetwerk (radiodeel). Door de fysieke beperkingen van de bandbreedte en het aantal kanalen dat ter beschikking staat, blijft de transmissiesnelheid beperkt tot 8 kbit/s.

4 Gerelateerde functies

In deze paragraaf worden enkele functies behandeld die nauw verwant zijn met het principe van packet switching. We hebben ons beperkt tot een beschrijving van de belangrijkste kenmerken van deze functies. Voor een meer gedetailleerde beschrijving wordt een aantal literatuurverwijzingen gegeven.

4.1 Buffering

Een direct gevolg van het store-en-forward-principe is dat knooppunten in staat moeten zijn pakketten te *bufferen*. Er moet dus bufferruimte beschikbaar zijn en zowel de toewijzing als de vrijgave van bufferruimte moeten worden geregeld *(buffer-management)*.

Wanneer met een vaste pakketlengte wordt gewerkt, zijn deze mechanismen vrij eenvoudig, daar het beheer van de bufferruimte kan plaatsvinden volgens deze vaste pakketlengte. Variabele pakketlengte wordt onder andere gebruikt in veel lokale netwerken. Voor Ethernet kan de pakketlengte variëren tussen 512 en 1518 octetten. Voor variabele pakketlengte wordt het beheer beduidend complexer. De grootte en de plaats van vrije bufferruimte moeten worden geadministreerd en de toewijzing van bufferruimte van variabele lengte vereist een plaatsingsbeleid zoals 'best fit, first fit' [2]. Ook het beheer van de vrije bufferruimte is complexer. Complexe buffer-management-mechanismen zorgen voor een vertragingsfactor vanwege de tijd die nodig is om de lengte van een pakket te bepalen (door bijvoorbeeld de lengte-indicator in het pakket te lezen) en de tijd die nodig is om (een deel van) het pakket tijdelijk op te slaan voordat het wordt doorgeschakeld (zie ook paragraaf 5.1). Daarom beperken netwerken voor high-speed packet switching, zoals ATM-netwerken, zich veelal tot een vaste pakketlengte.

4.2 Foutdetectie en foutcorrectie

Noodzaak

Transmissiemedia zijn bronnen van fouten. Zo is de foutkans (*bit error rate*) voor een glasvezelverbinding in de orde van 10^{-9}. Dit betekent dat gemiddeld 1 op de 10^9 verstuurde bits wordt verminkt. Dit is tamelijk weinig en is te danken aan de goede kwaliteit van glasvezel. Voor een coaxkabel is de foutkans in de orde van 10^{-6} en voor een traditionele telefoonverbinding ongeveer 10^{-4}. Een vaak gebruikte waarde voor een acceptabele foutkans voor de transmissie van een enkele bit van zender naar bestemming is 10^{-10}. Dit houdt in dat de foutkans voor een gebruiker 10^{-10} is. Dit leidt tot de noodzaak voor *foutdetecterende* en *foutcorrigerende mechanismen*.

Foutdetectie en foutcorrectie

Bij circuit switching komt een eventuele verminking pas tot uiting bij de bestemming, maar bij packet switching kan deze verminking al onderweg worden gedetecteerd (en eventueel gecorrigeerd). Het principe komt erop neer dat een zender redundante informatie toevoegt aan het te versturen pakket. Deze informatie

stelt de ontvanger in staat verminkingen te detecteren en eventueel te corrigeren; de correctiemogelijkheid is afhankelijk van de extra informatie die wordt toegevoegd. Bij de eenvoudigste vorm van foutdetectie wordt een enkele bit (*pariteitsbit*) toegevoegd, waarmee wordt aangegeven of het aantal enen in het pakket even of oneven is.

Als de ontvanger een fout detecteert die hij niet kan corrigeren, dan vraagt hij door het versturen van een *negative acknowledgement* (negatief bevestigingsbericht) aan de zender een *hertransmissie*. Een variant hierop is het niet versturen van een bevestigingsbericht door de ontvanger, waarna het aflopen van een timer bij de zender voor een hertransmissie zorgt. Merk op dat de mogelijkheid tot hertransmissies ontstaat door het bufferingsmechanisme. De zender weet bij ontvangst van een negative acknowledgement dat hij een hertransmissie moet uitvoeren. Een protocol waarbij de detectie van fouten door de ontvanger leidt tot hertransmissies, staat bekend als *ARQ-protocol* (Automatic Repeat reQuest).

Volgnummers

Tussen een zender en een ontvanger kunnen diverse pakketten onderweg zijn. Om een ontvanger in staat te stellen hertransmissie van een bepaald pakket aan te vragen, worden pakketten van een *volgnummer* voorzien. We zullen zien dat deze volgnummers ook ter ondersteuning van andere functies noodzakelijk zijn. Het eenvoudigste protocol dat in staat is fouten te detecteren en hertransmissies uit te voeren is het Alternating Bit Protocol uit 1969. Dit protocol gebruikt alternerend de nummers 0 en 1 als volgnummers.

Verlies van berichten

Pakketten kunnen niet alleen worden verminkt, ze kunnen ook verloren gaan. Dat gebeurt bijvoorbeeld als het bestemmingsadres in een pakket wordt verminkt, waardoor het pakket niet op de juiste bestemming aankomt. Om het verlies van pakketten te kunnen detecteren, wordt gebruik gemaakt van *timers*. Bij het versturen van een pakket zet de zender een lokale wekker op een bepaalde waarde. Als de wekker afgaat voordat een bevestiging van ontvangst (verzonden door de ontvanger) door de zender is ontvangen, is dit een indicatie dat het bericht verloren is gegaan. De zender gaat daarop tot een hertransmissie van het bericht over. Dit proces wordt herhaald totdat een bevestiging van (correcte) ontvangst wordt ontvangen. Deze bevestigingsberichten worden *acknowledgements* genoemd.

Voorbeeld

Ter illustratie van een foutcorrigerend mechanisme geven we een korte toepassing van een eenvoudige coderingstechniek met *Hamming-codering*. Deze techniek wordt in LAN's zelden toegepast, maar vindt zijn toepassing vooral in satellietsystemen. Bij Hamming-codering worden de bitposities in een gecodeerd octet genummerd van 1 tot en met $m+c$ (m is het aantal controlebits en c het aantal bits in de te coderen eenheid,

in ons geval een octet). Het i-de toegevoegde bit (controlebit) bevindt zich op bitpositie 2^i. De controlebits zijn zodanig in het gecodeerde octet geplaatst dat de som van hun posities in dit octet precies de positie aangeeft van de eventuele foute bit (in het geval van een enkele bitfout). Andersom geldt dat door de positie van een bit te schrijven als een som van machten van 2 de posities van de geassocieerde controlebits wordt verkregen. Bijvoorbeeld: $5 = 4+1$ geeft aan dat voor bit 5 de controlebits op posities 1 en 4 betrekking hebben.

Stel nu dat we het octet
100*0*100
willen versturen. Gecodeerd wordt dit
011000*0*1100
(de onderstreepte bits zijn de toegevoegde controlebits). Neem aan dat er een enkele bitfout optreedt en dat bit 7 wordt verminkt, zodat
011000*1*1100
wordt ontvangen. De ontvanger denkt dat het octet
100*1*100
is verstuurd, rekent hiervoor de controlebits uit en merkt dat ze verschillen van de ontvangen controlebits op de posities 1, 2 en 4. Dit betekent dat bit $1+2+4 = 7$ verminkt is ontvangen. Nu is de ontvanger (zonder hertransmissie van de zender) in staat bit 7 te wijzigen en aldus het originele bericht te reconstrueren.

4.3 Segmentatie en hereniging

Voor het transporteren van informatie in pakketten zijn twee functies nodig (zie figuur 3): een functie die de berichten splitst in pakketten (*segmentatie*) en de inverse functie die pakketten omzet in een stroom berichten (*hereniging* of reassembly). Deze functies zijn niet alleen in het beginpunt en het eindpunt aanwezig, maar doordat onderweg met verschillende pakketlengten kan worden gewerkt, kunnen ze ook in tussenpunten nodig zijn. Hierbij kan worden gedacht aan verschillende netwerken die onderling zijn verbonden door routers of bridges (bijvoorbeeld een ATM-netwerk met pakketten van 53 octetten, een Ethernet-netwerk met pakketten van maximaal 1518 octetten en een FDDI-netwerk met pakketten van maximaal 4500 octetten).

Volgorde herstellen

Door hertransmissies of doordat pakketten verschillende routes kunnen volgen, worden verzonden pakketten niet altijd in dezelfde volgorde ontvangen. Bij hereniging van de pakketten moet dus ook de oorspronkelijke volgorde van de pakketten worden hersteld. Hiervoor nummert de zender de pakketten opeenvolgend (modulo een bepaald maximum) en kan de ontvanger met behulp van deze volgnummers de oorspronkelijke volgorde herstellen. Merk op dat een dergelijk mechanisme niet nodig is bij circuitgeschakelde netwerken, omdat deze de volgorde van te transporteren berichten bewaren.

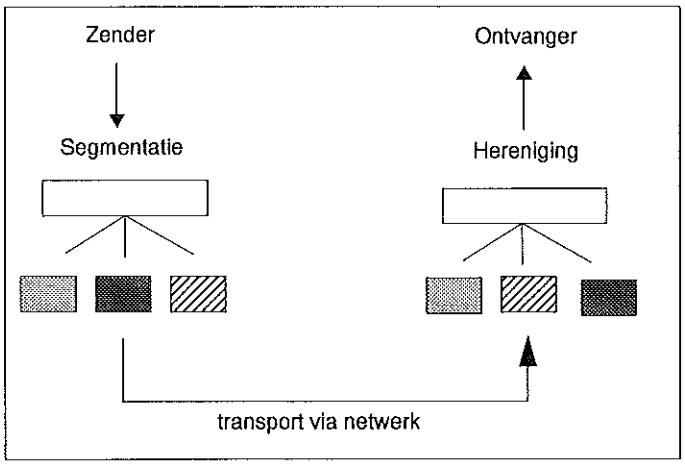

Fig. 3.
Segmentatie en hereniging.

4.4 Routering

Bij circuit switching wordt, voordat de uitwisseling van gebruikersinformatie plaatsvindt, een verbinding opgebouwd. Alle uit te wisselen informatie wordt vervolgens via deze verbinding gerouteerd. De routering (het bepalen van de route van zender naar ontvanger) vindt dus slechts een maal plaats, namelijk bij het opzetten van de verbinding. Bij packet switching ontstaat door de afwezigheid van a priori gereserveerde middelen de vrijheid om pakketten onafhankelijk van elkaar door het netwerk te routeren. Dit bevordert de foutbestendigheid van het netwerk: wanneer een bepaalde verbinding uitvalt (of van een zeer slechte kwaliteit is of overbelast is), kan een andere verbinding worden gekozen om een pakket verder te versturen (zie figuur 4).

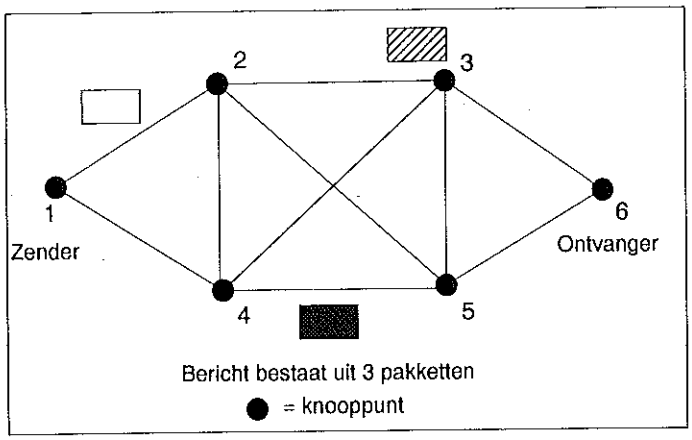

Fig. 4.
Routering van pakketten in een pakketgeschakeld netwerk.

Routeringsalgoritmen

De keuze van de volgende verbinding waarover het pakket zal worden verstuurd, wordt bepaald door een *routeringsalgoritme*. In computernetwerken worden verschillende routeringsalgoritmen toegepast. Het belangrijkste onderscheid in deze diversiteit van algoritmen is het onderscheid tussen *statische* en *dynamische* algoritmen. Bij statische algoritmen zijn de routeringsbeslissingen van tevoren berekend en ze blijven onveranderd zolang het netwerk operationeel is. Hierdoor wordt geen rekening gehouden met bijvoorbeeld het tijdelijk overbelast zijn of uitvallen van een bepaald knooppunt of een bepaalde verbinding. Bij dynamische routeringsalgoritmen is daarentegen de routeringsbeslissing afhankelijk van gegevens over het huidige verkeer in het netwerk. Met behulp van deze (complexere) algoritmen kan nu beter worden geanticipeerd op variaties in de intensiteit van het netwerkverkeer. Er moet worden vermeld dat ter ondersteuning van deze dynamische routeringsalgoritmen metingen van de verkeersintensiteit (of beschikbaarheid) van verbindingen moeten worden uitgevoerd, dat deze informatie moet worden verspreid over de tussenknooppunten en dat uiteindelijk in deze knooppunten de routeringsbeslissingen moeten worden aangepast aan de nieuwe situatie.

Eén van de bekendste (statische) routeringsalgoritmen is Dijkstra's *kortste pad*-algoritme, dat bij een gegeven topologie van het netwerk en de kosten van iedere verbinding (bijvoorbeeld de lengte van een verbinding of foutbestendigheid) bepaalt wat de kortste route is van ieder knooppunt naar ieder ander knooppunt. Voor een uitgebreid overzicht van routeringsalgoritmen wordt verwezen naar [3].

4.5 Flow Control

Noodzaak

Een belangrijke consequentie van het niet a priori reserveren van allerlei systeemmiddelen kan leiden tot overbelasting van een knooppunt. Dit betekent dat de ontvanger geen berichten meer kan opslaan (gebrek aan bufferruimte), bijvoorbeeld doordat de zender pakketten te snel achter elkaar verstuurt of doordat de ontvanger pakketten ontvangt over een verbinding met een grotere capaciteit dan de uitgaande verbinding. Protocollen die dit probleem verhelpen, heten *flow control*-protocollen.

Basisprotocollen voor flow control

Het eenvoudigste protocol voor flow control, het *Stop-en-Wacht*-protocol, vereist dat de ontvangst van ieder pakket door de ontvanger aan de zender kenbaar wordt gemaakt. Dit gebeurt door het versturen van bevestigingsberichten (*acknowledgements*). De zender mag pas een volgend pakket versturen als hij voor het vorige verstuurde pakket een acknowledgement heeft ontvangen. Zo'n acknowledgement heeft in feite een dubbele rol: het geeft aan dat de ontvanger het vorige bericht heeft ontvangen (en gebufferd) en geeft de zender toestemming voor het versturen van een nieuw pakket.

Doordat een zender steeds moet wachten op een bevestiging van de ontvanger, treedt een behoorlijk verlies van efficiëntie op. In een satellietsysteem met een verbinding van 50 kbit/s, een pakketlengte van 1000 bits en een tweewegpropagatietijd van 500 msec kost het versturen van een pakket 20 msec. Aannemende dat de ontvanger zeer snel het pakket kan ontvangen en verwerken, moet de zender 500 msec wachten voordat het volgende pakket mag worden verstuurd. De zender staat dus 96% (!) van de tijd te wachten. Om deze inefficiëntie tegen te gaan, wordt vaak vastgelegd dat een vast aantal (*W*) pakketten van zender naar ontvanger onderweg mag zijn. Pakketten kunnen nu apart door de ontvanger worden bevestigd of de ontvanger kan na een aantal correct ontvangen pakketten een bevestigingsbericht versturen (*block acknowledgement*). Flow control-protocollen die zijn gebaseerd op dit principe worden *sliding window*-protocollen genoemd.

Variaties

Er zijn veel variaties van flow control-protocollen in gebruik. Vaak zijn dergelijke protocollen uitgerust met hertransmissies, foutdetecterende mechanismen en nummering van pakketten om het uit volgorde geraken van pakketten of het verlies (of verminking) van pakketten te kunnen detecteren. De belangrijkste twee categorieën protocollen zijn de flow control-protocollen *selective repeat* en *go-back-N*.
Bij een selective repeat-protocol kan ieder willekeurig pakket dat tot een negative acknowledgement of tot het aflopen van de timer leidt, opnieuw worden verstuurd, onafhankelijk van andere pakketten.
Bij go-back-N-protocollen worden alle pakketten vanaf een bepaald volgnummer opnieuw verstuurd zodra de ontvanger aangeeft dat het pakket met dat volgnummer niet correct is ontvangen (het is dan niet meer van belang of de ontvanger pakketten met een hoger volgnummer al dan niet correct heeft ontvangen).

Het voert te ver om hier een uitgebreide beschrijving te geven van flow control-protocollen. Zie [4] en [5] voor een uitgebreid overzicht en een vergelijking van de verschillende protocollen.

4.6 Congestion control

Er treedt *congestie* op in het netwerk wanneer een toename in de hoeveelheid verkeer leidt tot een afname van de throughput in het netwerk (doorvoersnelheid; het aantal pakketten dat per tijdseenheid wordt ontvangen). Het principe van congestie is weergegeven in figuur 5.

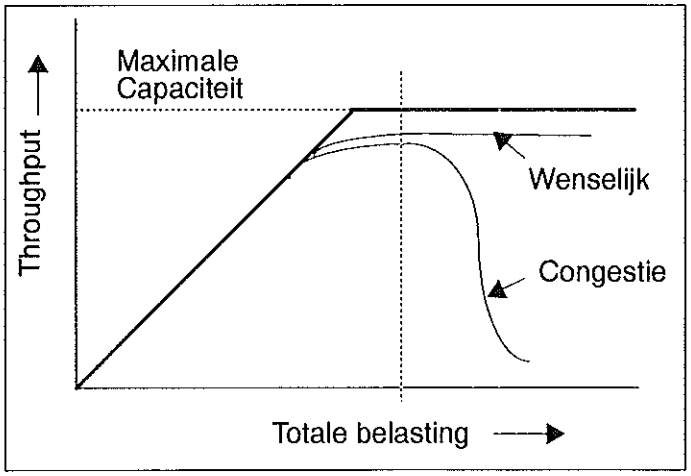

Fig. 5.
Netwerkcongestie.

In het ideale geval neemt de throughput rechtevenredig toe met de hoeveelheid aangeboden verkeer totdat de netwerkcapaciteit is bereikt. In de praktijk zal de optimale throughput niet samenvallen met de capaciteit van het netwerk, door de aanwezigheid van diverse protocolmechanismen (bijvoorbeeld door hertransmissies ten gevolge van verlies van pakketten). Wanneer de throughput optimaal is (aangegeven door de verticale stippellijn in figuur 5), leidt een toename van het verkeersaanbod tot een afname van de throughput.

Congestion control-protocollen hebben als functie de hoeveelheid verkeer te beheren, zodat er geen congestie kan optreden. Mogelijke technieken zijn het verwijderen van een pakket bij buffer-overflow in een tussenknooppunt of reservering van bufferruimte, zodat altijd een bepaalde hoeveelheid bufferruimte vrij is. Een andere oplossing is gebruik te maken van een flow control-protocol met een zich aanpassende venstergrootte, bijvoorbeeld: vergroot het venster met 1 bij ontvangst van elk bevestigingsbericht en verklein het venster met dezelfde hoeveelheid bij het afgaan van de timer. Hiervoor is de grootte van het venster (*window*) minimaal 1 en maximaal *W*.

4.7 Pakketstructuur

Nu een wat beter beeld is ontstaan van de extra functies die in een netwerk nodig zijn voor de ondersteuning van packet switching, wordt in deze paragraaf ter illustratie een typische pakketstructuur (Datalink-laag) toegelicht van een pakket met variabele lengte (zie figuur 6). Het eerste onderdeel van het pakket is een *preamble* en dient voor het aangeven van het begin van het pakket. Het *volgnummer* kan verschillende functies hebben, waaronder het mogelijk maken dat de ontvanger de oorspronkelijke volgorde van pakketten herstelt. Verder bevat

het pakket een identificatie van de *zender*, zodat de ontvanger weet van welk station het pakket afkomstig is. Dit adres kan vervolgens door de ontvanger worden gebruikt om te bepalen waar een eventuele acknowledgement naartoe moet. Het adres van de *bestemming* vermeldt de ontvanger en wordt gebruikt bij routering. Aangezien het een pakket met variabele lengte is, is er een onderdeel dat de *lengte* van de hoeveelheid databytes aangeeft. Deze lengte-indicator kan door de ontvanger worden gebruikt om te bepalen wat de laatste *data*byte van het pakket is. De *checksum* is extra informatie die door de zender is toegevoegd en die de ontvanger in staat stelt fouten te detecteren en te corrigeren.

Lengte (in octets)	*Functie van het veld*
1	preamble (geeft het begin van het pakket aan)
1	volgnummer van het pakket
2-6	adres van de zender
2-6	adres van de bestemming
1	lengte van het dataveld
0-25	dataveld
5	
2	checksum (extra informatie voor foutafhandeling)

Fig. 6. Een typisch voorbeeld van een pakketstructuur.

5 Kwantitatieve aspecten

In deze paragraaf wordt kort toegelicht wat de kwantitatieve aspecten van packet switching zijn. Achtereenvolgens komen aan bod de vertraging die pakketten kunnen ondervinden in het netwerk en de efficiëntie van het gebruik van pakketten als informatie-eenheid.

5.1 Doorgangstijd

Onder de *doorgangstijd* (of vertraging) die een bericht (dus niet een pakket) in een pakketgeschakeld netwerk kan ondergaan, verstaan we de tijdsduur tussen het verzenden van het bericht door de zender en de correcte ontvangst door de ontvanger van dit bericht. Deze doorgangstijd bestaat uit:

- *Transfertijd*
  De transfertijd bestaat uit twee componenten: de *transmissietijd* en de *propagatietijd*. De transmissietijd is de tijd die nodig is om een pakket te versturen en wordt hoofdzakelijk bepaald door de snelheid van het transmissiemedium en de pakketlengte. De propagatietijd is de tijd die nodig is om de afstand tussen twee knooppunten te overbruggen.
- *Segmentatietijd en herenigingstijd*
  De segmentatietijd is de vertraging die optreedt doordat het

bericht moet worden gesplitst in een aantal pakketten; de herenigingstijd is de vertraging ten gevolge van het inverse proces. Merk op dat deze vertragingen niet alleen optreden bij de zender en de bestemming, maar eventueel ook in tussenpunten waar een conversie moet plaatsvinden.

- *Vertraging door het schakelen*
  Deze vorm van vertraging bestaat hoofdzakelijk uit twee componenten. De eerste component is de *schakeltijd* die nodig is om (afgezien van buffering) een enkel pakket door het hele netwerk te schakelen. Deze vertraging is afhankelijk van de pakketlengte en het aantal tussenknooppunten en wordt hoofdzakelijk bepaald door de implementatietechnieken die zijn gebruikt. De tweede component is de *buffervertraging* die een pakket ondervindt doordat het telkens wordt gebufferd. Deze vertraging is afhankelijk van de verkeersintensiteit, de manier van bufferen (last-in-first-out of first-in-first-out) en de beschikbare buffergrootte.
- *Vertraging door hertransmissies*
  Het aantal hertransmissies van een pakket vormt ook een belangrijke component van de totale vertraging die een bericht kan ondervinden. Deze vertraging hangt af van de kwaliteit van de verbinding en van de toegepaste foutdetecterende en -corrigerende mechanismen.

Om een indruk te geven van de grootte van de verschillende geïdentificeerde componenten, nemen we het volgende voorbeeld [1].
Beschouw een ATM-netwerk met een propagatietijd van 4 msec/km. De afstand tussen zender en ontvanger is 1000 km, resulterend in een transmissietijd van 4 msec. Neem aan dat er twee tussenpunten zijn en dat de schakeltijd van een enkel tussenpunt 450 msec bedraagt (de gemiddelde waarde die het CCITT adviseert). Typische waarden bij een transfer- en schakelsnelheid van 150 Mbit/s en een pakketlengte van 64 bits zijn: een segmentatie- en herenigingstijd van 8 msec (voor ondersteuning van spraakoverdracht met 64 kbit/s), een buffervertraging van 800 msec en een schakeltijd van 900 msec. Doordat gebruik wordt gemaakt van glasvezelverbindingen, vinden er geen hertransmissies plaats.

5.2 Efficiëntie

Met efficiëntie wordt eigenlijk *pakketefficiëntie* bedoeld. Dat wil zeggen: de hoeveelheid werkelijke informatie in een pakket ten opzichte van de hoeveelheid overhead-informatie in dat pakket. Men moet bij overhead-informatie denken aan (voor de gebruiker irrelevante) informatie zoals adressen, routeringsinformatie, extra informatie voor foutdetectie en -correctie, en volgnummers. Deze overhead wordt voornamelijk veroorzaakt door het principe van packet switching en is in veel mindere mate aanwezig in circuitgeschakelde netwerken.

Voor pakketten van vaste lengte met een hoeveelheid informatie *I* (in octetten), een hoeveelheid overhead-informatie *H* (in octetten) en een hoeveelheid data *D* (in octetten) die tussen zender en ontvanger moet worden uitgewisseld, is de pakketefficiëntie:

$$E = \frac{D}{\left\lceil \frac{D}{I} \right\rceil \cdot (I + H)}$$

Hierin is $\lceil x \rceil$ (voor willekeurige x) het kleinste gehele getal groter of gelijk x. In het optimale geval is de hoeveelheid uit te wisselen informatie *D* een veelvoud van de hoeveelheid informatie *I* in een pakket. Dan vindt er geen *fragmentatie* plaats (dat wil zeggen dat een gedeelte van de informatieruimte in een pakket moet worden opgevuld met lege informatie, omdat de hoeveelheid te transporteren informatie te klein is) en is het gebruik optimaal:

$$E = \frac{D}{D+H}$$

Voor pakketten met een variabele lengte geldt de laatste gelijkheid voor *E*. Hierbij moet echter worden opgemerkt dat vaak extra (overhead-)informatie aan een pakket met variabele lengte wordt toegevoegd, zoals een indicatie van de lengte van het pakket. *H* zal dus groter zijn voor variabele pakketlengte.

In figuur 7 is de pakketefficiëntie *E* uitgezet tegen de hoeveelheid uit te wisselen informatie *D*. *I* is 48 octetten groot en *H* is 6 octetten. Voor de variabele pakketlengte is een additionele overhead van 2 octetten aangenomen voor de pakketlengte-indicatie in een pakket.

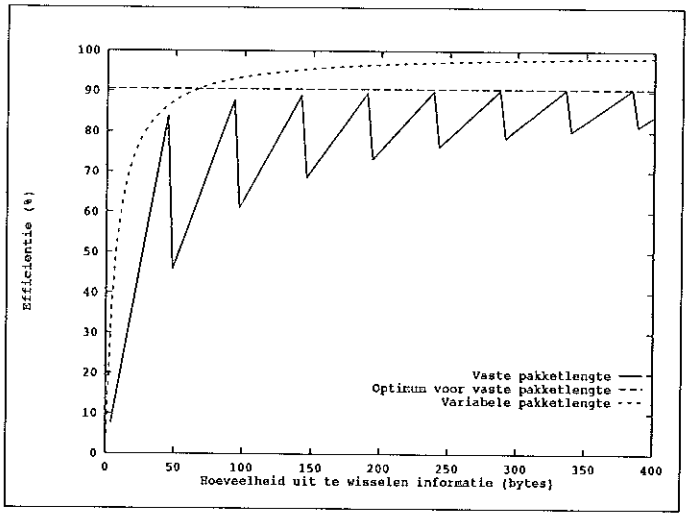

Fig. 7. Pakketefficiëntie (in procenten) versus de hoeveelheid te versturen informatie (in octetten).

5.3 Foutafhandelingsmechanismen

In paragraaf 3 is vermeld dat packet switching geschikt is gemaakt voor netwerken met grote bandbreedte door vereenvoudiging van de protocollen. Eén van de protocollen die is vereenvoudigd, is het error control-protocol. Door de verbeterde kwaliteit van transmissiemiddelen is het niet altijd noodzakelijk error control uit te voeren voor iedere verbinding tussen twee knooppunten (link-by-link error control), maar is het voldoende om alleen tussen zender en bestemming een dergelijk protocol te onderhouden (end-to-end error control).

In deze paragraaf bespreken we kort wat de invloed is van deze twee vormen van flow control op de hoeveelheid verkeer in het netwerk. Hierbij gaan we ervan uit dat er sprake is van een sliding window-protocol met een venstergrootte *W*, gebruik van volgnummers en hertransmissies bij foutdetectie. Wanneer de ontvanger aangeeft dat een bericht met een bepaald volgnummer niet correct is ontvangen, worden alle pakketten vanaf dat volgnummer die de zender al had verstuurd opnieuw verstuurd (go-back-N flow control). Aannemende dat er gemiddeld *W*/2 pakketten tussen zender en ontvanger onderweg zijn, wordt de hoeveelheid extra verkeer die door dit protocol wordt veroorzaakt, gegeven door:

$$V = \frac{W}{2} \cdot \frac{1 - (1-B)^{nL}}{(1-B)^{nL}}$$

Hierin is n het aantal verbindingen tussen zender en bestemming, L de lengte van het pakket (in bits) en B de bit error rate van de verbinding.

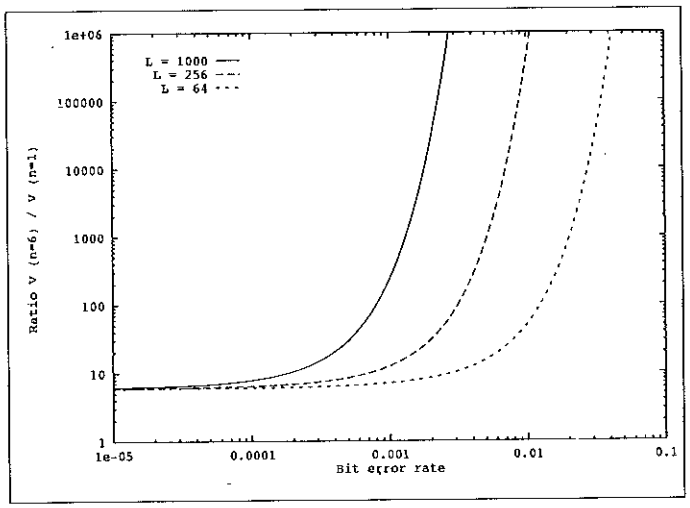

Fig. 8. Toename van de verkeersintensiteit voor link-by-link (n = 6) en end-to-end (n = 1) flow control versus de bit error rate.

Het is interessant te analyseren wat voor een gegeven pakketlengte het verschil is tussen de hoeveelheid geïntroduceerd verkeer bij link-by-link en bij end-to-end flow control. In figuur 8 is de ratio van V voor $n = 1$ (end-to-end) en $n = 6$ (link-by-link) weergegeven als functie van de bit error rate. Als venstergrootte is 10 pakketten genomen. Uit de figuur leiden we af dat voor een bitrate kleiner dan 10^{-4} de hoeveelheid verkeer bij link-by-link ten opzichte van end-to-end flow control nagenoeg voor alle pakketlengten gelijk is (V, 6). Dit betekent dat voor dergelijke bitrates het niet noodzakelijk is om link-by-link flow control toe te passen, maar dat end-to-end control flow voldoende is. Dit leidt tot een vereenvoudiging van de benodigde protocolondersteuning. Verder leiden we af dat voor kleinere pakketlengten dit zelfs voor een grotere bit error rate geldt. Dus hoe kleiner de pakketten, des te eerder de vereenvoudiging tot end-to-end flow control kan worden gemaakt.

6 Standaardisatie

Een van de belangrijkste aanbevelingen op het gebied van packet switching is ongetwijfeld de CCITT-aanbeveling *X.25*. X.25 is eigenlijk een hele familie van protocollen (X.21 is een onderdeel ervan), ondersteunt verbindingsgeoriënteerde services en wordt in heel veel openbare datanetwerken gebruikt, zoals het Nederlandse Unidata Datanet 1.

Een verbindingsloos netwerkprotocol dat in het begin van de jaren tachtig is ontwikkeld en wordt toegepast in het bekende ARPANET, is het *Internet Protocol* (IP). Dit protocol is, vaak samen met het daarop ontwikkelde *Transmission Control Protocol* (TCP), veel toegepast in tegenwoordige netwerken. IP bevat mechanismen om pakketten te routeren door het netwerk en mechanismen voor flow control, foutafhandeling, enzovoort.

Een verbindingsloze variant van X.25 is door OSI gestandaardiseerd als norm ISO 8473. Deze standaard staat beter bekend onder de naam *ConnectionLess Network Protocol* (CLNP) en is gebaseerd op IP. Dit protocol wordt onder andere toegepast in de Netwerklaag van MAP (Manufacturing Automation Protocol) en TOP (Technical and Office Protocol). Voor de onderliggende Datalink-laag wordt vaak een standaard uit de IEEE 802.X-serie (ofwel ISO 8802/X-serie) gebruikt. Een variant van dit protocol wordt ook toegepast in ISDN-netwerken (CCITT I.415).

De genoemde standaarden hebben alle betrekking op het leveren van een netwerkservice. Standaarden voor de onderste lagen (die dikwijls in verband worden gebracht met packet switching-technieken en die vaak een onderdeel vormen van de

genoemde standaarden) zijn te vinden in de familie van HDLC-protocollen (High-level Data Link Control), zoals de ISO-normen 3309, 4335, 7776, 7809 en 8471.

7 Literatuur

[1] M. de Prycker, *Asynchronous Transfer Mode: solution for broadband ISDN*, Ellis Horwood, New York 1991.
[2] A. M. Lister, *Inleiding Besturingssystemen*, Academic Service, 1985.
[3] M. Schwartz & T. E. Stern, *Routing Techniques used in Computer Communication Networks*, IEEE Transactions on Communication, vol. 28, no. 4, 1980.
[4] M. Gerla & L. Kleinrock, *Flow Control: A Comparative Survey*, IEEE Transactions on Communications, vol. 28, no. 4, 1980.
[5] G. J. Holzmann, *Design and validation of computer protocols*, Prentice-Hall, New York 1991.

Auteurs

De heer Katoen is werkzaam bij de Universiteit Twente, vakgroep Tele-informatica en Open Systemen. Hij is vooral betrokken bij de ontwikkeling van derde generatie mobiele netwerken in Europees verband.
De heer Van de Lagemaat is werkzaam bij de Universiteit Twente als instituutmanager van het Centre for Telematics en Information Technology (CTIT). Het CTIT is een multi-disciplinair onderzoeksinstituut op het gebied van ontwerp, invoering en gebruik van telematica-systemen en infrastructuren.